1989年3月1日発行(毎月1回1日発行)第8巻3号通巻83号　昭和58年11月2日第三種郵便物認可
ISSN 0910-7614

パーソナルコンピュータ・マガジン
MZシリーズ,X1/turbo,X68000&ポケコン

Oh!X

オー!エックス 定価540円

特集
BASIC“おもちゃ箱”

S-OS全機種共通システム
浮動小数点演算パッケージ

X68000デバイスドライバ
高速版FLOAT2+

X1/turbo
MZ-700版スペハリをX1で
ロボットシミュレーションTAMA

X68000
マシン語プログラミング講座(予告編)

THE SOFTOUCH
新九玉伝/ウォーニング
X68000用X1エミュレータ
Musicstudio PRO-68K

LIVE in '89
X1/turbo ANGEL
X68000 Moonlight Feels Right/ソーサリアン
MZ-2500 スペースハリアー・メインテーマ

猫とコンピュータ/知能機械概論
OS-9/X68000入門/C調言語講座PRO-68K

3
MAR. 1989

20MBハードディスクモデル

X68000
PERSONAL WORKSTATION
ACE HD

通商産業省選定

■本体＋キーボード＋マウス・トラックボール
CZ-611C-GY(グレー)・-BK(ブラック)標準価格399,800円
写真はCZ-611C-GY＋CZ-601D-GY＋CZ-6ST1-E

■本体＋キーボード＋マウス・トラックボール
CZ-601C-GY(グレー)・-BK(ブラック)標準価格319,800円
写真はCZ-601C-BK＋CZ-603D-BK

〈X68000ACEシリーズの主な特長〉●実装密度をさらに追求して信頼性を高めたマンハッタンシェイプ●68000搭載●テキスト、グラフィック、スプライト、独立3画面設計、最大12Mバイトの大容量メモリ(標準1Mバイト)●フレンドリーOS、Human 68k搭載●連文節変換、マルチフォントをサポートした強力日本語処理●1024×1024ドット(最大表示エリア768×512ドット)の実画面エリアを装備した高解像度表示能力●512×512ドット、65,536色同時発色●水平32、1画面128のパワフルなスプライト機能●オーバースキャン機能を採用した512×512ドットレベルのスーパーインポーズ●テキストビットマップ方式採用●8重和音ステレオFM音源搭載●音声デジタイズ記録AD PCM※採用●マウス・トラックボール標準装備●5インチ1MバイトFDD2基搭載●「X-BASIC」、「辞書ディスク」と各種ユーティリティ、「日本語ワードプロセッサ」をバンドル●20Mバイトハードディスク内蔵(CZ-611C)

※Adaptive Differential PCM

■15型カラーディスプレイテレビ(ドットピッチ0.39mm)CZ-601D-GY(グレー)・-BK(ブラック)標準価格119,800円
■15型カラーディスプレイテレビ(ドットピッチ0.31mm)CZ-611D-GY(グレー)・-BK(ブラック)標準価格145,000円
■14型カラーディスプレイ(ドットピッチ0.31mm)CZ-603D-GY(グレー)・-BK(ブラック) 標準価格84,800円(チルトスタンド同梱)
■チルトスタンドCZ-6ST1-E(グレー)・-B(ブラック)標準価格5,800円(CZ-601D/611D用)

資料請求券
X68000
oh!X
3係

アートの領域へ。

クォリティを維持しつづけることは、ある意味では創造することより困難なこととも言われています。出会いが印象的であればあるほど、その後が大変です。このことは、そのままX68000の歩みを言い得ているかも知れません。確かに技術は日進月歩です。しかしそれだけでコンピュータがもつべき創造性を論ずることはできないのも、また事実です。私たちはテクノロジーとクリエイティブマインド、いわば人とマシンとのソフトウェアインターフェイスで応えます。ホリゾンタルなマシンとしての熟成。そこからはいくつもの分野が見えてくるはずです。そしてどんな分野にしろX68000の仕事はアートであるべきです——。ますます洗練されて信頼性を高めたACEシリーズの登場で、あなたはまた新たな可能性に出会えそうです。

豊富な周辺機器がクリエイティブワークをサポート。

品名	型名	価格
●21型カラーディスプレイ	CU-21CD	標準価格139,800円
●RGBシステムチューナー	CZ-6TU	標準価格 35,800円
●カラーイメージスキャナ※1	CZ-8NS1	標準価格188,000円
●カラーイメージユニット※2	CZ-6VT1	標準価格 69,800円
●カラービデオプリンタ	CZ-6PV1	標準価格198,000円
●24ピン漢字プリンタ(80桁)	CZ-8PK7	標準価格122,000円
●24ピン漢字プリンタ(136桁)	CZ-8PK8	標準価格152,000円
●24ピン漢字プリンタ(80桁)	CZ-8PK9	標準価格 89,800円
●熱転写カラー漢字プリンタ	CZ-8PC3	標準価格 65,800円
●熱転写カラー漢字プリンタ	CZ-8PC2	標準価格 69,800円
●ハードディスクユニット(20MB)	CZ-620H	標準価格178,000円
●モデムユニット※3	CZ-8TM2	標準価格 49,800円
●RS-232Cケーブル(平行接続型)	CZ-8LM1	標準価格 7,200円
●RS-232Cケーブル(クロス接続型)	CZ-8LM2	標準価格 7,200円
●拡張I/Oボックス(4スロット)	CZ-6EB1	標準価格 88,000円
●1MB増設RAMボード(内蔵用)	CZ-6BE1A	標準価格 38,000円
●2MB増設RAMボード※4	CZ-6BE2	標準価格 79,800円
●4MB増設RAMボード※4	CZ-6BE4	標準価格138,000円
●FAXボード	CZ-6BC1	標準価格 79,800円
●MIDIボード	CZ-6BM1	標準価格 26,800円
●GP-IBボード	CZ-6BG1	標準価格 59,800円
●ユニバーサルI/Oボード	CZ-6BU1	標準価格 39,800円
●増設用RS-232Cボード(2チャンネル)	CZ-6BF1	標準価格 49,800円
●数値演算プロセッサボード	CZ-6BP1	標準価格 79,800円
●スキャナ用パラレルボード	CZ-6BN1	標準価格 29,800円
●システムラック	CZ-6SD1	標準価格 44,800円
●アンプ内蔵スピーカーシステム(2本1組)	AN-160SP	標準価格 59,800円
●マウス	CZ-8NM2A	標準価格 6,800円
●トラックボール	CZ-8NT1	標準価格 13,800円
●ジョイカード	CZ-8NJ1	標準価格 1,700円
●高性能CRTフィルター	BF-68PRO	標準価格 19,800円

※1 使用に際しては、カラーイメージスキャナCZ-8NS1に同梱のRS-232Cケーブルで接続するか、より高速のパラレルデータ転送を行う場合、別売のスキャナ用パラレルボードCZ-6BN1で接続してください。※2 使用に際してはコンピュータ本体と専用15型カラーディスプレイテレビ(CZ-601D、CZ-611Dなど)が必要です。※3 モデムユニットCZ-8TM2に同梱のソフトはX1/X1 turboシリーズ用です。※4 使用に際しては、あらかじめ、別売の1MB増設RAMボードCZ-6BE1Aを増設してください。

アートツールと呼びたい「PRO-68K」シリーズソフト。

内容	品名	型名	価格
イージーオペレーションの統合型表計算ソフト	BUSINESS PRO-68K	CZ-212BS	標準価格 68,000円
コマンド型リレーショナルデータベース	DATA PRO-68K	CZ-220BS	標準価格 58,000円
ワープロ機能を備えたカード型リレーショナルデータベース	CARD PRO-68K	CZ-226BS	標準価格 29,800円
FM音源をフルサポートするサウンドエディタ	SOUND PRO-68K	CZ-214MS	標準価格 15,800円
マウスを使った簡単操作の楽譜ワープロ	MUSIC PRO-68K	CZ-213MS	標準価格 18,800円
AD PCM機能をサポートしたサンプリングエディタ	Sampling PRO-68K	CZ-215MS	標準価格 17,800円
オリジナリティを活かせるポップアートツール	NEW Printshop PRO-68K	CZ-221HS	標準価格 19,800円
スクリーンエディタ内蔵の通信ソフト	Communication PRO-68K	CZ-223CS	標準価格 19,800円
ソフトウェア開発に役立つCコンパイラ	C compiler PRO-68K	CZ-211LS	標準価格 39,800円
24トラックのMIDIマルチレコーディングソフト	Musicstudio PRO-68K	CZ-237MS	標準価格 25,800円
MIDI楽器演奏が楽しめるMUSIC PRO-68KのMIDI版	MUSIC PRO-68K〔MIDI〕	CZ-247MS	2月発売予定
ソフトウェア開発ツール	THE 福袋 V2.0	CZ-224LS	標準価格 9,980円
マルチタスク、リアルタイムオペレーティングシステム	OS-9/X68000	CZ-219SS	標準価格 29,800円
本格財務会計ソフトウェア	TOP財務会計	CZ-227BS	標準価格200,000円
AI開発ツール	AI-68K(Staff LISP/OPS PRO-68K)	CZ-234LS	標準価格188,000円

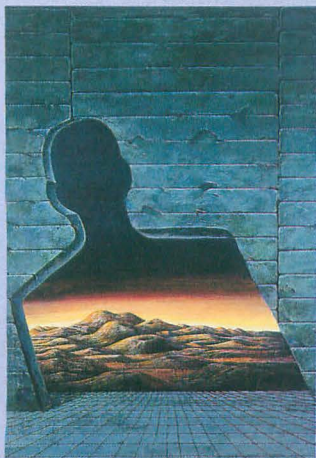
ゲームソフト	型名	価格
●ツインビー	CZ-217AS	標準価格 7,800円
●アルカノイド	CZ-222AS	標準価格 7,800円
●沙羅曼蛇	CZ-218AS	標準価格 8,800円
●熱血高校ドッジボール部	CZ-232AS	標準価格 7,800円
●フルスロットル	CZ-231AS	標準価格 8,800円

シャープパソコンフォーラム'89 in 赤坂/3月25日(土)・26日(日)/ラフォーレミュージアム赤坂
(赤坂ツインタワービル1F)
●詳しくは169頁をご覧ください。

シャープ株式会社
●お問い合わせは…シャープ㈱電子機器事業本部システム機器営業部 〒545 大阪市阿倍野区長池町22番22号 ☎(06)621-1221(大代表)
電子機器事業本部テレビ事業部第4商品企画部 〒162 東京都新宿区市谷八幡町8番地 ☎(03)260-1161(大代表)

Oh!X

CONT

〈スタッフ〉
●編集長／前田 徹 ●副編集長／永野 仁 ●編集／植木章夫 石塚康世 高野庸一 ●協力／有田隆也 中森 章 清水和人 後藤貴行 林 一樹 浅野恵造 山村 一 井本 泰 堀内保秀 荻窪 圭 藤原和典 岡本浩一郎 毛内俊行 野中俊一郎 吉田賢司 影山裕昭 相馬英智 古村 聡 村田敏幸 倉持亮一 ●カメラ／杉山和美 ●イラスト／永沢しげる 山田晴久 小栗由香 ●アートディレクター／島村勝頼 ●レイアウト／元木昌子 AD GREEN ●校正／手塚喜美子 千野延明

表紙絵：Matsubaguchi Tadao

UNIXはAT&T BELL LABORATORIESのOS名です。
CP/M,P-CP/M,CP/M Plus, CP/M-86,CP/M-68K,
CP/M-8000, C-DOSはDIGITAL RESEARCH
XENIX,MS-DOS, Macro 80,MS-OS/2はMICROSOFT
OS/2はIBM
SONY FilerはSONY
MSX-DOSはアスキー
S1-OSはMULTISOLUTIONS
OS-9, OS-9/68000はMICROWARE
UCSD p-systemはカリフォルニア大学理事会
FLEXはTSC
Word Star, Word MasterはMICRO PRO
TURBO PASCAL, SidekickはBORLAND INTERNATIONAL
LSI CはLSI JAPAN
HuBASICはハドソンソフト
SUPER BASE, WICSはキャリーラボ
の登録商標です。その他プログラム名，CPU名は一般に各メーカーの登録商標です。本文中では，“®”，“TM”マークは明記していません。
本誌に掲載されたすべてのプログラムは著作権法上，個人で使用するほかは無断複製することを禁じられています。

■広告目次

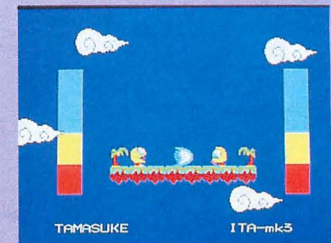
1989 MAR.

3

E N T S

特集「超簡単アニメーション技法」

特集　ブロックテニスで反則攻撃

特集「がんばれ！　カズシゲ君」

コミカルロボットゲームTAMA

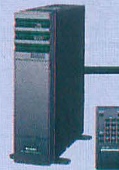
新九玉伝

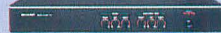
Musicstudio PRO-68K

クリエイティブマインド

21型カラーディスプレイ NEW
CU-21CD
標準価格 139,800円

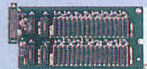
RGBシステムチューナー NEW
CZ-6TU-GY・-BK
標準価格 35,800円
(リモコン付)

表示

カラーイメージユニット
CZ-6VT1
CZ-6VT1-BK
標準価格 69,800円

映像入力

カラーイメージスキャナ※1
CZ-8NS1
標準価格 188,000円

スキャナ用パラレルボード
CZ-6BN1
標準価格 29,800円

画像入力

1MB増設RAMボード
(CZ-600C内蔵用)
CZ-6BE1
標準価格 35,000円

1MB増設RAMボード
(CZ-601C/611C内蔵用)
CZ-6BE1A
標準価格 38,000円

内蔵メモリ

● 写真はCZ-601C/CZ-603Dです。

2MB増設RAMボード※3
CZ-6BE2
標準価格 79,800円

4MB増設RAMボード※3
CZ-6BE4
標準価格 138,000円

ユニバーサルI/Oボード
CZ-6BU1
標準価格 39,800円

GP-IBボード
CZ-6BG1
標準価格 59,800円

増設用RS-232Cボード
(2チャンネル)
CZ-6BF1
標準価格 49,800円

数値演算プロセッサボード
CZ-6BP1
標準価格 79,800円

FAXボード NEW
CZ-6BC1
標準価格 79,800円

MIDIボード NEW
CZ-6BM1
標準価格 26,800円

拡張ボード

● 写真はCZ-611C/CZ-601D/CZ-6ST1です。

● 本体＋キーボード＋マウス・トラックボール
CZ-600C-E・-B　標準価格 369,000円
CZ-601C-GY・-BK　標準価格 319,800円
CZ-611C-GY・-BK　標準価格 399,800円

● 15型カラーディスプレイテレビ
CZ-600D-E・-B　標準価格 129,800円
CZ-601D-GY・-BK　標準価格 119,800円
CZ-611D-GY・-BK　標準価格 145,000円

● 14型カラーディスプレイ
CZ-603D-GY・-BK　標準価格 84,800円 NEW
(チルトスタンド同梱)

● チルトスタンド
CZ-6ST1-E・-B　標準価格 5,800円
(CZ-600D/601D/611D用)

● 高性能CRTフィルター
BF-68PRO　標準価格 19,800円 NEW
(CZ-600D/601D/611D/603D/CU-15M1用)

拡張I/Oボックス(4スロット)
CZ-6EB1
CZ-6EB1-BK
標準価格 88,000円

拡張スロット

※1 ご使用に際しては、カラーイメージスキャナ CZ-8NS1に同梱のRS-232Cケーブルで接続するか、より高速のパラレルデータ伝送を行う場合、別売のスキャナ用パラレルボードCZ-6BN1' 標準価格29,800円で接続してください。
CZ-6BE1標準価格35,000円(CZ-600C)、CZ-6BE1A 標準価格38,000円(CZ-601C、CZ-611C)を増設してください。

シャープ株式会社 ●お問い合わせは…… シャープ㈱電子機器事業本部システム機器営業部 〒545 大阪市阿倍野区長池町22番22号 ☎(06)621-1221(大代表)

資料請求券
X68000ペリフェラル
oh!X
3係

思わず熱くなる。

あふれる周辺機器がX68000をサポート。

シャープペリフェラルファミリー X68000

※2 モデムユニットCZ-8TM2に同梱のソフトはX1/X1ターボシリーズ用です。 ※3 ご使用に際しては、あらかじめ別売の1MB増設RAMボード

X1・X1 turbo シリーズ用周辺機器

品名	品番	価格
カラーディスプレイ		
●21型カラーディスプレイ※1	CU-21CD	139,800円
映像・画像入力編集装置		
●カラーイメージスキャナ	CZ-8NS1	188,000円
●カラーイメージボードII	CZ-8BV2	39,800円
●立体映像セット	CZ-8BR1	29,800円
●パーソナルテロッパ※2	CZ-8DT2	44,800円
FM音源		
●ステレオタイプFM音源ボード	CZ-8BS1	23,800円
スピーカー(2本1組)標準装備、ミュージックツール同梱		
プリンタ		
●24ピン漢字プリンタ(80桁)	CZ-8PK7	122,000円
●24ピン漢字プリンタ(136桁)	CZ-8PK8	152,000円
●24ピン漢字プリンタ(80桁)	CZ-8PK9	89,800円
●熱転写カラー漢字プリンタ	CZ-8PC3	65,800円
●熱転写カラー漢字プリンタ	CZ-8PC2	69,800円
●カラービデオプリンタ	CZ-6PV1	198,000円
ファイル		
●ミニフロッピーディスクユニット(2HD・2D)※3	CZ-520F	118,000円
●ミニフロッピーディスクユニット(2D)	CZ-502F	99,800円
●ミニフロッピーディスクユニット(2D・1ドライブ)	CZ-503F	49,800円
●増設用ミニフロッピーディスクドライブ(2D)※4	CZ-53F-BK	19,800円
●ハードディスクユニット※3	CZ-500H	348,000円
●ハードディスクユニット(増設用)	CZ-501H	258,000円
●カセットデータレコーダ	CZ-8RL1	24,800円
●ミニフロッピーディスク	CZ-5M2D/CZ-5M2HD(各10枚入)	
●コンパクトフロッピーディスク	CZ-3FBD	1,300円
拡張ボード・その他		
●モデムユニット(300/1200ボー)	CZ-8TM2	49,800円
●320KB外部メモリ	CZ-8BE2	29,800円
●ROM BASICボード※5	CZ-8RB	19,800円
●RS-232C・マウスボード※6	CZ-8BM2	19,800円
●フロッピーディスクインターフェイス※7	CZ-8BF1	14,800円
●JIS第1水準漢字ROM※8	CZ-8BK2	19,800円
●JIS第2水準漢字ROM※9	CZ-8BK4	6,800円
●JIS第2水準漢字ROM＆ターボ博士レキシコン・日本語百科ワードパワー※10	CZ-8BK3	13,800円
●RS-232C用ケーブル(平行接続型)	CZ-8LM1	7,200円
●RS-232C用ケーブル(クロス接続型)	CZ-8LM2	7,200円
●拡張I/Oボックス	CZ-8EB3	33,800円
●RFコンバータ※11	AN-58C	2,980円
●マウス	CZ-8NM2A	6,800円
●トラックボール	CZ-8NT1	13,800円
●ジョイカード	CZ-8NJ1	1,700円
●チルトスタンド※12	CZ-6ST1-E・-B	5,800円
●高性能CRTフィルター	BF-68PRO	19,800円
●システムスタンド	CZ-8SS2	5,500円
●スキャナ用パラレルボード※13	CZ-8BN1	27,800円

(価格は標準価格です。)

●品番中の-表示は、B〈ブラック〉・E〈オフィスグレー〉を示します。※1 X1ターボZシリーズ用 ※2 CZ-862Cには接続できません ※3 X1ターボシリーズ用 ※4 CZ-830C用 ※5 X1シリーズ用V1.0 ※6 X1シリーズ用 ※7 CZ-850CでCZ-520Fを使用する場合に必要 ※8 CZ-800C、801C、802C、803C、811C、820C用 ※9 CZ-856C用 ※10 CZ-850C、851C、852C、862C用 ※11 CZ-820C、822C、830C用 ※12 CZ-600D、601D、611D、880D、830D、CU-15M1用 ※13 CZ-8NS1用 ●接続等の説明につきましては、周辺機器総合カタログをご参照ください。

電子機器事業本部テレビ事業部第4商品企画部〒162東京都新宿区市谷八幡町8番地☎(03)260-1161(大代表)

プロ指向のワークステーションをお届けします。

X68000上のソフト開発を強力にサポート

C&プロフェッショナル・パッケージ

XCP576　¥58,000

OS-9/X68000上で動作するマイクロウェア・CコンパイラV3.1とユーティリティ・ソフトのパッケージです。動作させるためにはOS-9/X68000が必要です。

マイクロウェア・Cコンパイラは、
K&RとANSIに準拠したスタンダードC ライブラリ群と
X68000のために拡張した豊富なライブラリ群とをサポートした
強力なCコンパイラです。X68000のために拡張されたライブラリとしては　●フロッピー／ハードディスク　●日本語処理　●ADPCM　●FM音源　●マウス　●ジョイスティック　●グラフィクス　●PSS（プレゼンテーション・サポート・システム）があります。

パッケージの内容

■ソフトウェア　●Cコンパイラ　●標準ライブラリ　●OS-9/X68000専用ライブラリ　●PSS（プレゼンテーション・サポート・システム）サポートライブラリ　●ヘッダファイル　●マクロアセンブラR68　●リンカL68　●MAKEコマンド　●ユーザステイト・シンボリック・デバッガDEBUG　●漢字フル・スクリーンエディタKμMACS　●フル・スクリーンエディタSCRED

■マニュアル　●Cコンパイラ・ユーザーズ・マニュアル　●付属ユーティリティ・マニュアル　●OS-9/X68000専用ライブラリ・マニュアル

microware®
マイクロウェア・ジャパン株式会社
〒273 千葉県船橋市本町4-41-19 Phone：0474-22-1747

開発効率の向上と操作性の統一!! Finalで実現できます。

マルチファイル・スクリーン・エディタ「Final」のX68000版をご紹介します。

Finalは複数のファイルを同時に編集することを目的に設計されたスクリーン・エディタです。マルチファイル機能とウィンドウ機能を使って、効率の良いプログラム開発を行なうことができます。また、高速化されたカーソル移動や画面表示スピードと、ポップアップ・メニューのサポートなどユーザ・フレンドリな編集環境を提供しています。

Finalは、現在15機種のMS-DOSマシン、さらにOS／2、X-windw上でも稼動しています。X68000版Finalの登場により、統一された操作環境がまた一歩広がりました。プログラム開発環境の統一……それがFinalのコンセプトです。

Final X68000 マルチファイル・スクリーン・エディタ

X68000版Finalの特徴

- X68000版日本語フロントプロセッサ「ASK68K」に対応
- X68000の画面サイズ(96カラムX30行)に対応
- 22文字までのファイル名に対応
- 画面表示しない文字列の置換モードを追加
- ファイル行数が999, 999行まで対応
- 画面背景色、文字色等で65,000色の中から任意の色を選択可能
- キーリピート開始の時間設定が可能
- ヒストリー読み込みの場合のステップ入力が可能

Finalの機能

●マルチ・ファイル

最大で10個までのファイルを、同時に編集することができます。オープンされているファイルの中から、2つのファイルをウインドウに呼び出し、編集することができます。

●文字列検索・置換

文字列の検索および置換では、正規表現とワイルドカードの使用、大文字と小文字を識別しないモード……などがサポートされています。

●フリー・カーソル機能

編集画面上の任意の位置にカーソルを最短距離で移動する、フリー・カーソル・モードがサポートされています。このモードを使ったテキスト編集では、キャリッジ・リターンの後ろにもカーソルを移動することが可能です。

●ワンタッチ短縮入力

キーに割り当てられた登録文字列、つまりC言語であれば「oblcvalg(&arg・n-prt)n-prt」のようなステートメントや「unsignedchar」などの予約語などを、ワンタッチでカーソル位置に書き込むことができます。

●ビルトインgrep

外部ユーティリティーとして提供されるgrepをFinal内部から実行することで、その検索結果を編集ウインドウにオープンできます。カーソルでジャンプ先を選んでタグ・ジャンプ・コマンドを実行すると、ファイルをオープンし、該当する行へカーソルを移動します。

●コマンド・カスタマイズ

初期設定ではWordstarライクなキー設定が行われていますが、全てのキー割り当ての再設定が可能です。好みのキー・アサインで、Finalを操作できます。

●その他の機能

ポップアップメニュー、バイナリー・エディット機能、オンメモリー・エディットウインドウ拡大と縮少、同一ファイルのウインドウ分割、ウインドウの縦横分割、カット＆ペースト、BOX型カット＆ペースト、ヘルプ画面の表示、ヒストリー、フロファイル、オート・インデント、キーボード・マクロ、パーマネント・マクロ、タグ・ジャンプ、行のマークとジャンプ、子プロセス実行、指定範囲のファイル書き出し、カーソル位置へのファイル読み込み、対応カッコの検索、ASCIIコード表、コントロール・コードの入力……etc.

Finalの対応機種

機種	価格	機種	価格
●PC-9801シリーズ ●PC-98LT ●PC-286 ●X-68000	￥38,000	●PC-9801＋VJE-β ●PC-98XA/XL ●FM-16βシリーズ ●FMR-30/50/60/70 ●HITACHI/B16シリーズ ●HITACHI/2020 ●IBM55シリーズ ●リコーマイツール ●J3100/3300シリーズ ●M500/M700/M800 ●PC-286＋VJE-β ●AXパソコン	￥48,000

サイトライセンスについて

㈱エー・エス・ピーでは、企業、大学および教育研究機関のためのサイト・ライセンス契約を実施しています。このサイト・ライセンス契約は、ソフトウェアを大量導入するユーザーに対して、低価格でご利用して頂くために設けられたライセンス・システムです。

1. 契約本数内のハードウェア上で使用可能です。
2. 契約本数の追加変更も可能です。
3. バージョンアップ・サービスを無償で受けられます。
4. マニュアル等は必要部数を実費にて購入できます。

今、Finalをお買上げいただきますと、もれなくFinalの単行本をプレゼント!!

期間　1989年2月より4月末日
対応機種　PC-9801シリーズと他機種（上記の機種）
対象メディア　5"2HD/2DD版　3.5"2HD/2DD版

日本人のためのソフトウェアを創造する
株式会社 エー・エス・ピー
〒143 東京都大田区大森北5-8-11-106
TEL.03(767)1451　FAX.03(767)1453

とことんレイ・トレソフト！カンタン

君はもう買ったか売れてるCG

★告知 学校教材用としてサイクロンを使ってみませんか？サンプルソフトお分けします。

★アニメキット(5,000円)近日発売！

サイクロンは、レイ・トレソフトの常識です。

X68000

3Dレイ・トレーシングCGツール **サイクロン68K**

58,000円

★Z'S STAFF(PRO68K)とそのままやりとりできます。

● 彩croneモデラー　● 彩croneレンダラー

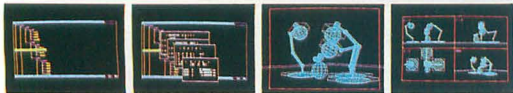
X68000で気軽に3次元CGを楽しみたい方へ

※高機能アニメーションソフト開発中！

※各メーカーの周辺機器(アウトプット)へのインターフェースソフト完備。プレゼンテーションの世界が広がります。

■効能

■建築デザイン/外観パース、室内パース、インテリア、エクステリア、室内レイアウト　■環境設計/都市景観シミュレーション、環境シミュレーション、■グラフィックデザイン/ポスター、チラシ、ロゴタイプ、シンボルマーク、パッケージ　■広報/会社案内ビデオ、会議用スライド、教育ビデオ、プレゼンテーション・アニメ　■工業デザイン/商品開発、製品シミュレーション　■芸術/ビデオアートなど映像ビジネス全般に効果が有ります。

ここが違う、サイクロン!! ーモデリングがすべてを決める！ー

- 他では考えられない簡単入力。
- ANS独自の階層構造型モデラーで、リスト記述はいっさい不要。
- 4面ワイヤーフレームでビジュアル形状確認。(3面図、パース図)
- サイクロンだけ!!　論理演算で(　)が使える。

アニメーションも可能!! 強力なマクロ編集機能

- 単体プリミティブの色・質感まとめて変更。
- 物体の配置変更自由自在。
- 結局部品が生きるライブラリー。

サイクロンの主な取扱い店

■北海道地区(札幌)
そうご電器YES5F
ツクモ札幌1号店

■東北地区(仙台)
電巧堂Dac東口店
庄司デンキコンピューター中央

■東京地区
石丸電気マイコンセンター
ツクモ7号店
ツクモNC店
ツクモ5号店
ラオックス本店
ラオックス中央店
ラオックスサウンド
ラオックスニューサウンド
ロケット本店
ロケット2号店
ロケット3号店
ロケット5号店
ミナミ電気館
ヤマギワソフトショップ
ヤマギワテクニカ
ラオックス新宿5F
CSK西口
サンゴーカメラ
西武池袋店
J&P渋谷店
ソフトクリエイト渋谷店
新橋パソコンセンター
J&P八王子そごう店
ソフトクリエイト聖蹟桜丘店
ムラウチ
ラオックス吉祥寺
J&P町田
ラオックス三鷹

■関東地区(神奈川)
ラオックス厚木
野島will
ソフトクリエイト横浜
IC WORLD TOTSUKA

■関東地区(茨・群・栃)
パソコンランド21前橋
パソコンランド21高崎

■東海地区
メルバ静岡
すみやパソコンランド
メルバ沼津
メルバ浜松店
浜松すみやパソコンランド

■名古屋地区
栄電社テクノ
マイコンテック

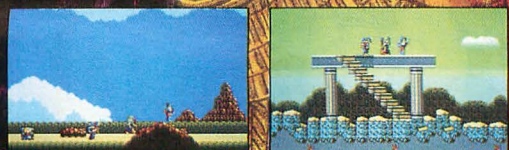
■大阪地区
J&Pメディアランド
J&Pビジネスランド
J&Pテクノランド
J&Pコスモランド
上新電機阪急三番街
マイコンショップCSK
ニノミヤパソコンランド難波
ニノミヤパソコンランド日本橋
ニノミヤパソコンランドえびす
ニノミヤパソコンランド駅前
ニノミヤエレランド

■近畿地区
J&P京都寺町
上新電機さんのみやいち番館
栄電社三宮Cスペース
J&P和歌山

■中国・四国地区
ダイイチパソコンシティ

■九州地区(福岡)
ベストマイコン福岡店
カホマイコンセンター

株式会社アンス・コンサルタンツ　九州本社/〒810 福岡市中央区平丘町68
phone. 092-522-6347　FAX 092-521-0400

Falcom
迫り来る“魔”の波動を感じたか!?
ならば旅立て、
暗黒のギルバレス島へ!!
SORCERIAN SYSTEM
SCENARIO
Vol.3
ピラミッド ソーサリアン
Sorcerian System
『戦国ソーサリアン』で暗黒神・邪鬼を倒すことは、“魔”の物語の序章にしかすぎなかった! 暗黒神・邪鬼を凌ぐ強大な“魔”の本体、大魔王ギルバレスに挑む冒険だから、『ピラミッド・ソーサリアン』が、『戦国ソーサリアン』以上のボリュームを持っているのは、当然かもしれないが、それにしてもすさまじい。5インチ2Dディスクでは、やっとのことで2枚に収めたものの、内容の濃さは5枚分!? 1から5までが連なった長編シナリオであるというだけでなく、謎解きあり、アクションありで、各シナリオごとに、その性格が変えられているんだ。これほど彩り豊かで、これほど手ごわいゲームなんて、『ソーサリアン』にもなかったぜ!
SORCERIAN SYSTEM SCENARIO Vol.3
Pyramid Sorcerian
ピラミッド ソーサリアン
好評発売中!
X1turbo 5.2D(2枚組)3,800円
SORCERIAN
Sorcerian System
SCENARIO Vol.2
戦国ソーサリアン
好評発売中!!
SORCERIAN SYSTEM
SCENARIO Vol.2
戦国ソーサリアン
Falcom 日本ファルコム株式会社
Personal Computer Software
〒190 東京都立川市柴崎町2-1-4 トミオービル
通信販売(送料無料)
●現金書留の場合
氏名・機種名・住所・氏名・電話番号を明記して、現金書留でお申し込みください。
●代金引換の場合
電話やFAXやハガキで、品名・機種名・住所・氏名・年齢・電話番号を明記して、お申し込みください。商品お届け時に商品代金をお支払いください。
TEL 0425(27)6501
FAX 0425(28)2714

スタッフ募集／グラフィック・プログラマー・シナリオライター・ゲームデザイナー(自宅作業可能)、及び作品の持ち込み歓迎します。

ScapTrust 株式会社スキャップトラスト
〒150 東京都渋谷区神宮前5-42-1
ユーザーサポートテレホンサービス(03)486 8127

これが僕のステイタス。

あれこれ迷っている人、すぐウェーブ・アイへTELして下さい。

お申し込みは料金無料のフリーダイヤルで

0120-009898

ウェーブ・アイ10(テン)ポイントチェック

チェック1 全品２倍保証！
メーカー保証の2倍の保証がついています。(メーカー1年保証＋ウェーブ・アイ1年保証)

チェック2 夏のボーナス一括払いOK！
商品は今すぐお手元に、お支払いはまとめて夏のボーナスで!!

チェック3 超低金利クレジット
3回〜72回までのクレジットが格安の金利でOK。また当社提示支払い例のほかにお客様独自の支払いプランが組めます。

チェック4 商品先取り、支払いは半年先から。
支払い開始は半年先！でも商品はほしいというお客様でもOK！

チェック5 ボーナス2回払いOK！
月々の支払いは全くナシ！お支払いは冬と夏のボーナスでOK！

チェック6 代金引換OK！
現金一括にしたいというお客様、お支払いは現品到着時でOK！(但し離島の方はご利用できません。)

チェック7 全国無料配送
一部地域を除き送料無料で商品をおとどけします（但し5万円以上の商品に限ります。離島の方は有料となります。）

チェック8 配達日指定OK！
留守がちの方の為に、ご都合に合わせて、配達致します。もちろん日曜・祭日もOK。

チェック9 下取り、買取りもOK!
お手持ちのパソコンを下取りして、わずかな予算で新製品と買い換えることができます。

チェック10 ハガキ注文もOK！
いそがしくて電話をするひまが無いという方の為に、ハガキでのご注文もOK!

〒252
藤沢市湘南台
一丁目一〇番地一号
ウェーブアイ
Oh！X係

1. 住所
2. 氏名
3. 年令
4. 電話番号
5. 保護者名 ㊞ (20才未満の方)
6. 商品名
7. 支払い方法
月々 円× 回
ボーナス 円× 回

X68000 ACE HD

X68000に20MBハードディスクを搭載。ますます熱くなるクリエイティブ&パーソナルワークステーション。

本体とディスプレイの基本セットも特価でご奉仕!! お電話でお問合わせ下さい。

プラン348 X68000ACE HD 本格セット TELにて

品名	価格
CZ-611C(20MBハードディスク内蔵)	399,800円
CZ-603D (0.31ミリ、高解像度CRT)	84,800円
CZ-8PC3(80桁、カラー熱転写プリンター)	65,800円
C compiler PRO68K(ソフトウェア開発Cコンパイラ)	39,800円
A4カット紙100枚	480円
ブランクディスケット3M(5インチ2HD)10枚	18,000円
定価合計	608,680円

クリーニングディスク・マウスパッドサービス

ウェーブ・アイ特価

月々	ボーナス
10,000円×36回	ボーナス29,400円×6回
7,000円×48回	ボーナス27,400円×8回
5,000円×60回	ボーナス27,900円×10回
8,300円×72回	ボーナス なし

プラン350 X68000ACE HD アートセット TELにて

品名	価格
CZ-611C	399,800円
CZ-603D(0.31ミリ、高解像度CRT)	84,800円
CZ-8PC3(80桁、カラー熱転写プリンター)	65,800円
Z'S STAFF-PRO68K(グラフィックツール)	58,000円
PRINT-SHOP-PRO68K(高機能ポップアートツール)	19,800円
CZ-6TU(RGBシステムチューナ)	35,800円
CZ-6VT1(カラーイメージユニット)	69,800円
A4カット紙100枚	480円
ブランクディスケット(5インチ2HD10枚)	18,000円
定価合計	752,280円

クリーニングディスク・マウスパッドサービス

ウェーブ・アイ特価

月々	ボーナス
12,000円×36回	ボーナス40,800円×6回
10,000円×48回	ボーナス27,600円×8回
8,000円×60回	ボーナス25,000円×10回
10,500円×72回	ボーナス なし

プラン351 X68000ACE HD ミュージックセット TELにて

品名	価格
CZ-611C	399,800円
CZ-601D(0.39mm、TV内蔵CRT)	119,800円
CZ-8PC3(80桁、カラー熱転写プリンター)	65,800円
MUSIC PRO-68K(楽譜ワープロ)	18,800円
SOUND PRO-68K(FM音源サウンドエディタ)	15,800円
ED-700(2段パソコンラック)	27,000円
A4カット紙100枚	480円
ブランクディスケット3M(5インチ2HD)10枚	18,000円
定価合計	665,480円

クリーニングディスク・マウスパッドサービス

ウェーブ・アイ特価

月々	ボーナス
12,000円×36回	ボーナス26,800円×6回
9,000円×48回	ボーナス22,700円×8回
7,000円×60回	ボーナス22,000円×10回
9,200円×72回	ボーナス なし

本体とディスプレイの基本セットもございます。詳しくはTELにて。

X68000 ACE

ハードの余裕がフレンドリーなオペレーションを生みだしている。ますます熱くなるクリエイティブワークステーション。

プラン345 X68000ACEワープロセット TELにて

品名	価格
CZ-601C	319,800円
CZ-603D (0.31ミリ、高解像度CRT)	84,800円
CZ-8PC3(80桁、カラー熱転写プリンター)	65,800円
EW(日本語ワープロソフト)	38,000円
ED-700(パソコンデスク)	27,000円
A4カット紙100枚	480円
ブランクディスケット3M(5インチ2HD)10枚	18,000円
定価合計	553,580円

クリーニングディスク・マウスパッドサービス

ウェーブ・アイ特価

月々	ボーナス
9,000円×36回	ボーナス25,500円×6回
6,000円×48回	ボーナス25,800円×8回
4,000円×60回	ボーナス27,500円×10回
7,400円×72回	ボーナス なし

プラン346 X68000ACEアートセット TELにて

品名	価格
CZ-601C	319,800円
CZ-603D (0.31ミリ、高解像度CRT)	84,800円
CZ-6TU(RGBシステムTVチューナー)	35,800円
CZ-8PC3(80桁、カラー熱転写プリンター)	65,800円
CZ-6VT1(カラーイメージユニット)	69,800円
Z'S STAFF PRO68K	58,000円
A4カット紙100枚	480円
ブランクディスケット3M(5インチ2HD)10枚	18,000円
定価合計	652,480円

クリーニングディスク・マウスパッドサービス

ウェーブ・アイ特価

月々	ボーナス
12,000円×36回	ボーナス25,300円×6回
9,000円×48回	ボーナス21,600円×8回
6,000円×60回	ボーナス27,000円×10回
9,000円×72回	ボーナス なし

プラン347 X68000ACEミュージックセット TELにて

品名	価格
CZ-601C	319,800円
CZ-603D(0.31ミリ、高解像度CRT)	84,800円
CZ-8PC3(80桁、カラー熱転写プリンター)	65,800円
Music PRO68K(簡単操作の楽譜ワープロ)	18,800円
Sound PRO68K(FM音源サウンドエディタ)	15,800円
Sampling PRO 68K(高機能サンプリングエディタ)	17,800円
AN-160SP(アンプ内蔵スピーカーシステム)	59,800円
A4カット紙100枚	480円
ブランクディスケット3M(5インチ2HD)10枚	18,000円
定価合計	601,080円

クリーニングディスク・マウスパッドサービス

ウェーブ・アイ特価

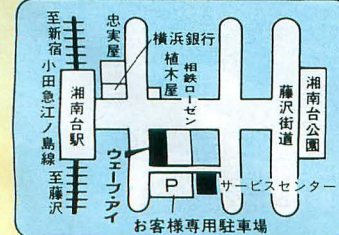
月々	ボーナス
10,000円×36回	ボーナス29,400円×6円
7,000円×48回	ボーナス27,400円×8回
5,000円×60回	ボーナス27,900円×10回
8,300円×72回	ボーナス なし

夏の 商品今すぐ!! ボーナス一括払いOK

受付時間 AM9:30▶PM9:00 電話一本即納！

地域	電話番号	地域	電話番号
藤沢	0466(43)1775	静岡	0542(54)0696
札幌	011(771)4971	名古屋	052(581)4325
盛岡	0196(24)3172	大阪	06(362)5057
仙台	022(267)5371	広島	082(293)0811
新潟	0252(75)5076	福岡	092(481)0502
東京	03(226)9286	FAX	0466(43)1265

18歳未満の方は保護者とご一緒にお電話下さい。

新歩人の情報ターミナル

湘南台店☎0466-43-1771

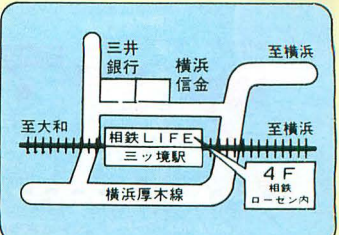
〒252神奈川県藤沢市湘南台1−10−1
振込銀行▶ 横浜銀行 湘南台支店 当座000467 (株)ウェーブ・アイ
第二・第三火曜日定休日

三ッ境店☎045-363-7044

ツクモならではのお知らせで〜す。

春のわんさかバザール

ソフト・ハードも新しい情報がわんさかどっさり。
3月は毎週ためになる説明会を行ないます。
●MIDIソフト ●OS-9/X68000 ●彩クロン
●Z's STAFF PRO-68K V2.0 etc……(於)7号店2F
★詳しくは7号店2F シャープコーナー(荒井)
までお問い合せください。

ツクモの X68は
買う前納得!
お値段ナットク!
買ってからなっとく!
の三拍子。

7号店は、外から見て2FにX68000の
シンボル「ツタンカーメン像」がめじるしです。

今、ハードディスクが売れている!

●アイテックハードディスク
IT X-203 (20MB 28ms)
ツクモ特価¥73,800
IT X-403 (40MB 29ms)
ツクモ特価¥104,800
(X-203/403はブラックかグレーをご指定下さい。)

予告
近日発売! TS-Series
3.5″増設ドライブ(X68000用)
●TS-3X68
●キーボード延長ケーブル
お問い合せは7号店まで……

プリンターはお持ちですか?

●カラー漢字熱転写プリンター
CZ-8PC3……定価¥65,800
●24ピン漢字プリンター(80桁)
CZ-8PK7……定価¥122,000
●24ピン漢字プリンター(136桁)
CZ-8PK8……定価¥152,000
●24ピン漢字プリンター(80桁)
CZ-8PK9……定価¥89,800
●カラーイメージジェットプリンター
IO-730……定価¥230,000

X68000 ファミリー好評発売中!

X68000 ACE HD CZ-611C (20MBハードディスク内蔵タイプ)……定価¥399,800
X68000 ACE CZ-601C (標準タイプ)……定価¥319,800

ディスプレイ

CU-21CD 21型カラーディスプレイ……定価¥139,800
CZ-601D ドットピッチ0.39ミリ……定価¥119,800
CZ-611D ドットピッチ0.31ミリ……定価¥145,000
CZ-603D ドットピッチ0.31ミリ……定価¥84,800
■オプション
CZ-6ST1 チルト台……定価¥5,800
CZ-6TU RGBシステムチューナー……定価¥35,800
BF-68PRO 14・15インチCRTフィルター……定価¥19,800

周辺機器

CZ-8BE1 1MB増設RAMボード(CZ-600C用)……定価¥35,000
CZ-6BE1A 1MB増設RAMボード(ACEシリーズ用)……定価¥38,000
CZ-6BE2 2MB増設RAMボード……定価¥79,800
CZ-6BE4 4MB増設RAMボード……定価¥138,000
CZ-6BC1 FAXボード……定価¥79,800
CZ-6BP1 数値演算プロセッサボード……定価¥79,800
CZ-6VT1 カラーイメージユニット……定価¥69,800
CZ-8NS1 カラーイメージスキャナー……定価¥188,000
CZ-6BM1 MIDIボード……定価¥26,800

豊富なソフトウェア

Kamikaze(神風) 統合型スプレッドシート……特価¥57,800
SOUND PRO-68K サウンドエディタ……定価¥15,800
MUSIC PRO-68K ミュージックツール……定価¥18,800
Sampling PRO-68K AD PCM活用ソフト……定価¥17,800
Musicstudio PRO-68K MIDIマルチレコーディングソフト……定価¥25,800
MUSIC PRO-68K[MIDI] MUSIC PRO-68KのMIDI版……近日発売
Communication PRO-68K 通信ソフト……定価¥19,800
DATA PRO-68K リレーショナルデータベース……定価¥58,000
CARD PRO-68K カード型データベース……定価¥29,800
Z's STAFF PRO-68K グラフィックツール……特価¥49,000
New Print Shop PRO-68K 高機能ポップアートツール……定価¥19,800
彩CRONE レイトレーシングソフトウェア……特価¥49,300
C COMPILER PRO-68K C言語開発セット……定価¥39,800
AI-68K AIプログラム開発ツール……定価¥188,000
OS-9/X68000……定価¥29,800
C & Profesional Package OS-9用C開発セット……定価¥58,000
C-TRACE68 レイトレーシングソフト……ツクモ特価¥57,800

その他、ビジネスソフト・ホビーソフトも多数発売中ですので、お気軽にお訪ねください。

X1 turbo Z III セット
●CZ-888B-BK……¥169,800
●CZ-860D-BK……¥99,800
ツクモ特価販売中
★上記セットに買い換えるなら

下取り機種	差額
CZ-852C+CZ-850D	¥172,000
CZ-856C+CZ-870D	¥170,000
CZ-822C+CZ-820D	¥190,000

X1 G モデル30セット
●CZ-822CB 本体
●CZ-820DB ディスプレイ
●人気ゲームソフト
●オリジナルゲームパック
●ディスケット……サービス
ツクモ特価¥79,800
※X1 twinも特別販売中ですヨ!

NEW Z-BASIC
X1ターボシリーズ対応 CZ-141SF
(64KB メモリーボード付属)
驚異の大特価! ¥9,800 (〒1,000)

MIDIセット
CZ-6BM1 MIDIボード……定価¥26,800
CZ-237MS Music Studio PRO-68K 定価¥25,800
MT-32 ローランドMIDIサウンドモジュール 定価¥69,000
ツクモ特価¥99,800
※Music PRO-68K(MIDI)とのセットはお問い合せ下さい。

モデム
オムロン
MD-2400B 300/1200/2400ボー……ツクモ特価¥39,800
MD-1200A2 300/1200ボー……ツクモ特価¥17,800

マウス/トラックボール
(X1、X1turboシリーズ、MZ-2500シリーズ対応)
シャープマウス
CZ-8NM2(A)……定価¥6,800
シャープトラックボール
CZ-8NT1……定価¥13,800

電子手帳もポケコンも!

シャープ
PA-8500
定価¥28,000
大型4行表示、データスケジュール管理に便利。ICカード、プリンタで更に発展するハイグレードタイプ。
特価 ¥24,800

シャープ PC-E200
定価¥22,000
特価¥17,800

シャープ PC-E500
定価¥28,800
特価¥24,800

ツクモは「スーパーX PRO SHOP」です。

秋葉原各店
至お茶の水
不忍通り
5号店
7号店
カート下御茶側 ニューセンター店
中央通り
JR秋葉原駅
(営)AM10時〜PM7時 (休)毎週木曜日

PRO STAFF ツクモ

ツクモ7号店 ☎03-253-4199
通信販売部 ☎03-251-9911
ツクモ5号店 ☎03-251-0531
ニューセンター店 ☎03-251-0987
名古屋1号店 ☎052-263-1655
名古屋2号店 ☎052-251-3399
ツクモ札幌 ☎011-241-2299

九十九電機(株) 〒101-91 東京都千代田区神田郵便局私書箱135号

全国代金引き換え配達
お申し込みは☎03-251-9911へお電話1本!
商品到着の際、玄関でお会計ができます。配達日の指定もできます。

夏・冬、ボーナス2回払い受付中
月々¥3,000以上の均等払いも頭金なし。

現金書留なら
〒101-91 東京都千代田区神田郵便局私書箱135号
九十九電機(株)通信販売部

銀行振込なら
事前に☎でお届け先をご連絡下さい。
富士銀行 神田支店(普)No.894047

警告の時はせまる!!

SFウォーシミュレーション+ロールプレイング

WARNING

ウォーニングTYPE 68

バトルモード

宇宙空間や惑星上では、行く手を阻む海賊やモンスター達と戦わなければならないことも。戦闘に必要な船の砲塔やロボット、アイテム等は、自分の戦術に合った物を購入できる。

移動モード

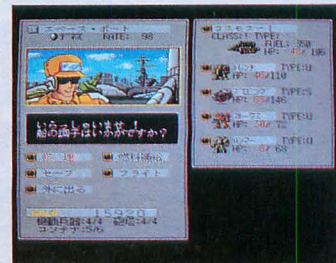
宇宙商人であるキミは、お金儲けのため、あるいは情報収集のため、星から星へと渡り歩く。着陸すれば、その星を駆け巡り、様々な場所や施設を訪れることができる。

★個性あふれる11タイプの宇宙船!
★様々な機動兵器と砲塔群により、自分の戦術に合った戦闘部隊が編成可能!
★個別目標指定・エネルギーチャージルール等、シミュレーション性豊かなバトルモード!

★プレイをスムーズにするレーダー&サテライトモード!
★貿易で資本金を増やし、装備をより強くしていくユニークなゲームシステム!

サテライト&レーダーモード

宇宙航行中、レーダーで位置の確認ができる。また地上ではサテライザーを打ち上げ、その星全体の様子を見わたすことができるので、目的地や各施設を発見しやすく、スムーズなプレイが可能。

★スピーディなゲーム展開が楽しめるスムーズな操作性!
★プレイを熱くさせるオリジナルBGMの数々!
★こっそり搭載された禁断の兵器や謎のアイテム!

ショップモード

マーケット、スペースポート、パーツセンター等、貿易や宇宙船の買い換えなどに、なくてはならない施設が数多く存在する。特にギルドでは、様々な情報をメッセージにより得ることができる。

大小7つの惑星と1つの人工惑星からなる『プシェード星系』。
そこではそれぞれの惑星が国家を築き、貿易を行っていた。
しかし『プシェード星系』は、多数のならず者たちがはびこる無法地帯でもあった。
キミは自分の宇宙船を駆り、遭遇した相手と時には取り引きし、時には戦わなければならない。
唸るキャノン砲! 砲火をくぐりぬける機動兵器!
あらゆる武器と希望を満載して、大宇宙を突き進むスペースシップ!
数々の謎を前に、キミならこれらをどう使いこなすか!?

好評発売中! ■X68000 2枚組 ¥7,800

未知との調和と創造
株式会社コスモス・コンピューター

〒150 東京都渋谷区桜ヶ丘29-24 秀和桜ヶ丘レジデンス611 ☎03(770)1821(代)

※仕様は改良のため予告なく変更することがあります。

通販

通信販売をご希望の方は、現金書留で当社までお申し込みください。送料は無料です。

資料請求券 Oh!X3

シートレース
C-TRACE

この画像はC-TRACEで作成したものです。(背景はスーパータブロー)

プロフェッショナル「CG作家」募集!!

★資格：C-TRACE登録ユーザーであること。★目的：職業作家としての自立を援助し、CGを広く世間一般に普及させる。★方法：C-TRACEにより作られたデータを当社のSYSTEM-100により高画質演算し、この作品をフォトライブラリーにて運用する。また、依頼制作も受注する。★費用：作家は一切を必要としません。データを送り、運用益を受取るだけです。演算料金、プリント費用、その他すべて当方で負担いたします。●資料請求は、切手を貼り宛先を記入した返信用封筒を同封の上、下記までお送り下さい。※C-TRACEユーザーは、シリアルNo.を明記して下さい。
〒158 東京都世田谷区等々力2-1-13 キャスト「プロ作家募集」係

■成長を続けるC-TRACEファミリーただいま予約受付中!!

①**C-TRACE98TP新発売**／トランスピュータを使用し、速度は10倍(対VX)。ソフト、ボードセット¥510,000。並列バージョン価格未定。②**C-TRACE68TP開発中**／③**C-TRACE98FB新発売**／C-TRACE98DRYとフレーム・バッファのセット。アナログRGBモニタに640×400Pix, 1670万色同時表示。テキスト・スーパーインポーズ　¥98,000。④**C-TRACE68FB開発中**／X68000が1670万色同時表示になる。
これら新製品は店頭販売いたしません。直接当社までお申し込みください。

■動作環境
PC-9801シリーズ全機種
RAM 640KB
MS-DOS Ver.2.11以上
コプロセッサー(8087,80287,80387)有無どちらも対応
■現在サポートしているフレームバッファ(X68000は本体のみ)
PC-9801　本体内VRAM
スーパーフレーム　サピエンス社
ハイパーフレーム　デジタルアーツ社

「C-TRACE68」は、演算スピードが3倍になりました!!

登録ユーザーには、バージョンアップいたします。
未登録の方は、大至急登録カードをお送りください。
バージョンアップの御案内をお送りいたします。

C-TRACE 98DRY (PC-9801対応)	¥68,000
C-TRACE 98+ (PC-9801対応)	¥198,000
C-TRACE NEWS (SONY)	¥380,000
C-TRACE 68 (X68000対応)	¥68,000

株式会社 キャスト
〒158 東京都世田谷区等々力2-1-13
TEL.03-705-0656 FAX.03-705-5224

Oh!X readers'ぎゃらりい

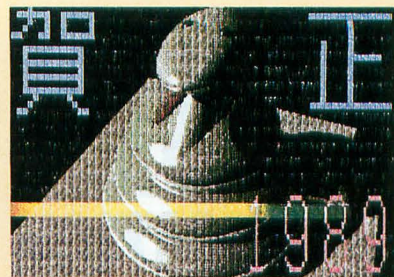
1989年度 カラーイラスト&年賀状

読者の皆さん，たくさんのイラスト入り年賀状をありがとうございました。今年は自粛かなぁなんて弱気になっていたOh!Xですが，これを紹介せねばパソコンに未来はないと思うほどの力作ぞろいとなりました。

▲笠井　清美　(北海道)

▲大和田　昭彦　(埼玉県)

▲清水　雅夫　(神奈川県)

▲上田　修　(三重県)

▲渡辺　久孝　(岡山県)

▲泉　広明　(福島県)

▲崎村　賢二　(東京都)

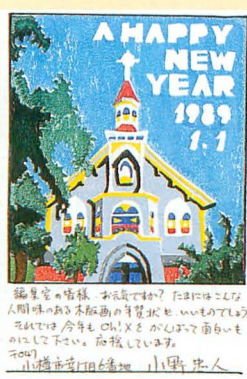
▲大津　和之　(福岡県)

▲丸藤　俊之　(神奈川県)

▲藪田　俊平　(和歌山県)

▶宮本　康司　(大阪府)

◀加藤　信夫　(宮城県)

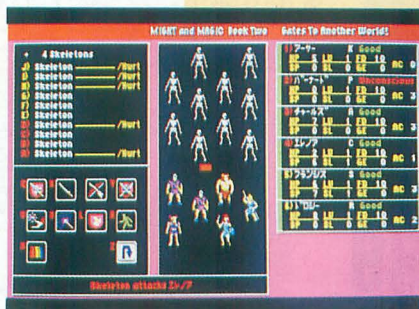
▲佐々木　伸一　(神奈川県)

▲小野　忠人　(北海道)

▲森下　保　(静岡県)

というわけで，「Oh!X readers'ぎゃらりい」は5月号の「言わせてくれなくちゃだワ」へと続くのでした。皆さんよろしくお願いします。なお，イラストに書かれていた電話番号の部分は，編集部の判断で修整させていただきましたのでご了承ください。

THE SOFTOUCH

SOFTWARE INFORMATION

信長の野望・戦国群雄伝
たんば
トリトーン2

おっとついに出ました，パックマニア。このゲームには，アーケードで熱中していたころの懐かしい思い出を持っている方は多いはず。さらに下の写真は，X1に発売され，間もなくX68000にも登場予定の大作RPG，Might & Magic IIなのです

話題のソフトウェア

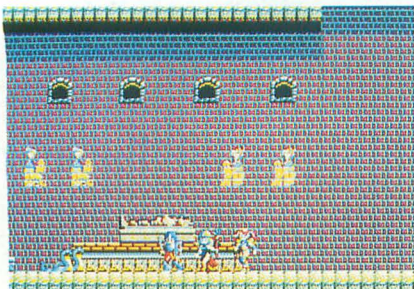
なぜか関東地方は，ポカポカと暖かい冬で毎日なにを着て外出すればいいのか迷ってしまうような日々が続いていますが，皆さん元気にゲームしてるでしょうか。

今月は，春をイメージできる最新作というわけで，いきなりパックマニアのオープニング画面をアップでお届けしてみました。どうです，この明るさ。とってもいいでしょ。これがカラフルな敵キャラ相手に，立体迷路をぴょこぴょこジャンプしながら動き回るのですから，楽しめることは請け合い。3月になれば店頭にも並ぶはずですから，楽しみにしていてください。

さて，来月はいよいよ1988年度 "GAME OF THE YEAR" の発表です。たくさんのおハガキをお送り頂いて，どうもありがとうございました。いま最終集計にとりかかっているところで，編集室はハガキの束を相手に大騒

読者が選ぶゲームベスト10

今月の結果を見てみると，対照的なのがソーサリアンとイースII。やはり，ソーサリアンはピラミッドソーサリアンが昨年末に発売されたせいか，余裕をみせながらの1位となりました。その代わりというか，息の長い人気を保っていたイースIIはジリジリと後退し，逆にX68000の最新作が上位に突如として登場してきています。特に琥珀色の遺言が先月10位だと思っていたら，なんと今月は入れ替わりでMurder Club DXが，いきなり2位に登場です。そのほかTETRISやサイオブレードといったところが注目株。春の新作ソフトの発売ラッシュを控えて，今後ますます混乱が予想されそうです。

1．ソーサリアン（追加シナリオ含む）
2．Murder Club DX
3．ドラゴンスピリット
4．めぞん一刻・完結編
5．TETRIS
6．ラスト・ハルマゲドン
7．サンダーフォースII
8．サイオブレード
9．信長の野望・全国版（X68000）
10．イースII

ぎ。それにしても，愛読者カードの記入欄には，ほとんどの方がそれぞれなんらかのゲーム名を記入してくれていることから考えてみると，質の高いゲームソフトが充実しているとこれだけ反響が違ってくるのかと改めて感心してしまいます。前年度なんかは，三国志かイース，そしてX68000のスペハリの3本くらいしか注目度の高いゲームってなかったわけですからね。

だから前回は，皆さんからのハガキもいまひとつパワーが感じられなくて，こちらとしても，品不足だったゲーム界を恨めしく思ったものです。

しかし，今年は違います。なにが違うかって？　わかりやすく数字で紹介すると，なんといっても，愛読者カードのGAME OF THE YEARに関する記入率が，なんと97%を占めたこと。つまり，ほとんどの方が記入してくれているのです。さらに，残りの3%はまったくの白紙かというと，そうではないのです。そこにはキチンと「受験だから，ゲームはやっていません」などの棄権理由が，ちゃーんと明記されているのです。これには担当者が涙，涙の大感激。さらにはこれまであまり関心を示していただけなかった30歳，40歳以上の方でも，嬉しいことにゲーム名と推薦理由をそれぞれご記入頂いているのです。

これだけ関心度が高ければ，あとはそれを集計して結果を出すだけで，きっと皆さんが満足していただける発表ができると，こちらもより一層気合が入ります。ほんとうに，どうもありがとうございました。と，締めくくるのにはまだ早い。もうひとつ，今年はなにが凄いかって，ハガキ全体のなかで**自由応募部門**に投票してくれた方の多いこと，多いこと。単純に割り出しても，ざっとハガキ全体の3割がこのノリなのですよ。それもゲームの悪口ばかり書いているわけじゃなくって，よくいえば「ウケ狙い」，悪くいえば「担当者が苦労してノミネート作品の集計を徹夜作業で仕上げたのにアッサリ無視して，よくもまあ，これだけ好き勝手なことばかり平気で書いてこれるもんだ。君たちいい根性してるじゃないか。ちょっとは人の話を真面目に聞きなさい」という内容のものばかりなのです。あとの説明が長くなったのはお許しください。なにを隠そう集計で徹夜したのはこの私なものですから，ついついホンネが……。

チラッとここでその中身をバラしてしまうと，「X68000がほしいで賞」とか，「なにがなんだか最初はわからなかったで賞」とか，「Oh!Xらしくていいで賞」，「本当にするとは誰も思っていなかったで賞」などなど，誰もそこからゲーム名が想像できないようなものばかり。なかにはハガキにあったカッコのなかに入りきらないものだから，2行や3行にも及ぶながーい名前を勝手に付けたハガキなんかが，嵐のように毎日舞い込んできたのです。「ゲッ，なにこれ。まさかこんなのばかりじゃないだろうな」と，担当者（この私）が不安になるのも，ご理解いただけるでしょ，この内容じゃ。

でも，ご安心ください。勝手に疑心暗鬼に陥っていた担当者(くどいようだが，この私)があさはかでした。各部門別の投票率を見てみると，Oh!X 大賞がダントツのトップ。続いてシューティングゲーム賞，Oh!68賞と並んでいるのです。やはり，皆さんは真剣に投票してくれていたのです。ホントにどうもありがとうございました。

さて，来月の目玉商品は，このGAME OF THE YEARの発表だけではありません。もうひとつ，ゲーム特集を大々的に予定しているのです。登場するゲームをここで簡単にご紹介しておくと，X1では**ウィザードリィ#4, サイオブレード，ピラミッドソーサリアン，今夜も朝までPOWERFULまあじゃん2**（こちらはX68000版もセットでご紹介），**リボルティーII**。一方のX68000関係では，果たして本当に4月号の特集までに間に合うか興味津津の**R-TYPE**と**大海令**。そして**ボスコニアン**(これは清水和人氏が挑戦します)，**SUPER大戦略68K，ソフトでハードな物語2，パワーリーグ，パックマニア**などなど，ズラッと並べただけでもなかなかに凄いメンバーでしょ。でも，なかには現在開発進行中のものもいくつか含まれていますから，もしそれまでに間に合わないものがあったとすれば，いまのうちに謝っておきます，ごめんなさい。このほか最新ゲーム情報のなかでは，HEシステム，メガドライブまで，幅広く紹介してみたいと思っていますので，大いに期待していてくださいね。

これら選りすぐりのゲームたちをいかに料理し，新しい趣向で応えられるか腕の見せどころとばかりにがんばるつもりですが，いまのところ決まっているテーマが，**「主役はいまここにいる」**なのです。

さてさて，なにがいったい飛び出すかは，4月号を見てのお楽しみ。結果をご覧になったあとは，ぜひ皆さんの感想をお聞かせください。それでは，また来月のゲーム特集でお会いしましょう。

新作ソフト情報

☆……1月30日現在発売中　★……近日発売予定

★信長の野望・戦国群雄伝

シミュレーションファンなら一度はプレイしたことがあるであろう，信長の野望。その全国版に続いて登場するのがこの「戦国群雄伝」だ。今回は，群雄割拠と信長の野望という2つのシナリオからなり，舞台となる領地は北海道，東北，九州を除く38カ国。そこで，プレイヤーは自分が選択した大名となり，国を治め，外交を行い，さらには領土拡大のための戦争をしかけていく。概要的にはまったく変化していないが，軍師の役割が重要なポイントとなったり，またその軍師を育てるための釆配を行うなど，多くの人材をいかに忠実で，能力の高い人物に育てるのかがゲームの勝敗に大きくかかわってくるようになっている。とにかく，200名以上の武将がしのぎを削る戦国での知力の勝負は，このシリーズならではの魅力を持っている。また，ゲームに使われたオリジナル曲を収めたCD付きサウンドウェアシリーズも同時発売される。

X1turbo用　5″2D版3枚組　9,800円
CD付きサウンドウェア　12,300円
（2ドライブ専用）
光栄　☎044(61)6861

たんば

☆たんば

相原コージと高橋章子の絶妙なコンビネーションでX68000ユーザーを楽しませてくれた，あの「たんば」が，ついにX1turboでも遊べるようになった。このゲームの正式名称は「霊界すごろく・たんば」といい，現世と霊界とをまたにかけ，輪廻転生を繰り返しながら，じりじりと神となるべく死後の世界の先に待ち受けるゴールに向かって一目散というストーリー。最初はミジンコやフジツボから始まって，ツチノコやコウモリ，スカンクへと徐徐にバージョンアップを重ね，さらには地獄界や霊界，天上界めぐりへと，ほとんど世紀末を象徴するようなこのゲーム。とにかく，ボードゲームがうまくパソコンとマッチしたといえる不思議な一作だ。

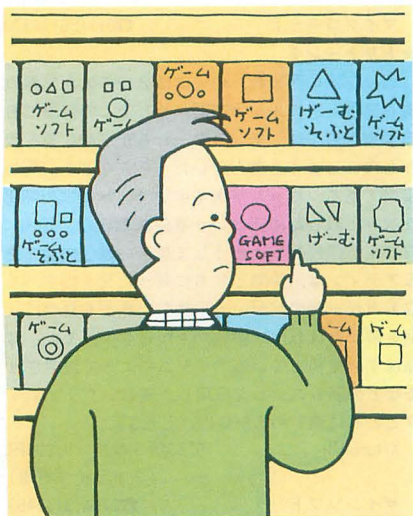
X1turbo用　5″2D版2枚組　6,800円
（2ドライブ専用）
マイクロネット　☎011(561)1370

★トリトーン2

周りを海に囲まれ，中央には2つの大きな湖のある島国，アムルシガ帝国。そしてこの帝国にある洞窟や森には，モンスターたちが平和な国を我がものにしようと潜んでいた。そんなとき，突如，皇帝であるシヴァが軍隊を率いて，無抵抗の国民を次々と殺し始めるという事件が起きた。恐怖におびえる国民たち。そうして皇帝の双子の弟であるアディンは，仲間とともに狂った兄を倒すべく立ち上がったのだが……。美しい島国を舞台に，謎を解かなければ経験値が上がらない，また，魔法はすべて持っているアイテムによって決定されるなどの新システムを採用し，美しいグラフィックとともに楽しめるRPGといえそうだ。

X1turbo用　5″2D版5枚組　9,800円
（2ドライブ専用）
ザインソフト　☎0794(31)7453

GAME REVIEW

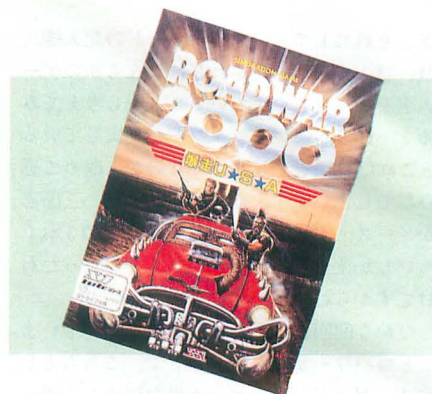
卒業・新入学シーズンを控えていますので，進路が決まってノンビリしている方，そうでない方，いろいろひっくるめて，比較的マイペースで遊べるシミュレーションとAVGをセットで今月はご用意してみました。ごゆっくりお楽しみください。

ロードウォー2000

細菌戦争によって崩壊したアメリカ大陸を舞台に，ギャング団が抗争を繰り広げる，新しいタイプのシミュレーションです。

▶部下を引き連れ車に乗り込み，アメリカを舞台に駆けまわります。目的は，バクテリアに冒された国民を救うため，8人の科学者を捜し出して研究所へ連れて行くという立派なものですが，なぜかわれらはギャング団。よって目的を遂行するために，武器や食料を略奪し，侵略軍などと流血の争いをし，ときには都市を支配したりして，本当に立派なのかどうだか（考えてみれば信長だって似たようなものですね）。

ゲームは基本的にシミュレーションタイプなのですが，マップは単純な方眼タイプだし，戦略というほど頭を使う必要はなさそうです。慣れてくると要領もつかめ，仲間や車も増強されて，どんどん略奪，ばんばん応戦しようと思うところへ，反応の鈍さが障害となってきます。マップの切り換えやウィンドウの開閉などが遅いのです。BGMもなく，セーブも1カ所でしかできませんが，やり込めば夢中になれる要素も持っているように思います。

熱中度▶▶▶▷▷▷▷ (お)

▶シミュレーション？　ウォーゲーム？　なんだかよくわからないゲームである。あえていうならば，経営シミュレーションなのだろう。とにかく，ギャング団のボスとなり，細菌戦および核戦争後のアメリカを救うのがこのゲームの目的である。マニュアルがわかりにくくて，ゲームの目的がわか

るまでひと苦労だった。そしてゲームを始めてからも，なにかしようとするたびに，わけのわからないマニュアルと格闘せねばならない。ついでにいうならば，表示のスピードがあまりに遅い。いくらなんでもひどすぎる。なんとかしてもらいたい。

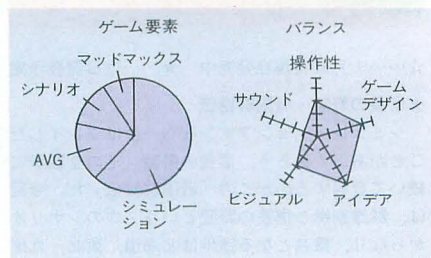
と，いいたいことをいってみたが，詰めの甘さとマニュアルの質と，プログラムの遅さがそれを損なっているとはいえ，アイデアとしてはよくできたゲームである。一見面白そうだが，やってみるといらいらする。しかし，実は面白いのでは？　という気もしないではない。なんとも不思議なゲームだ。

熱中度▶▶▶▶▷▷▷ (M.Y.)

X1/X1turbo用　5"2D版2枚組　7,800円（2ドライブ専用）
スタークラフト　☎03(988)2988

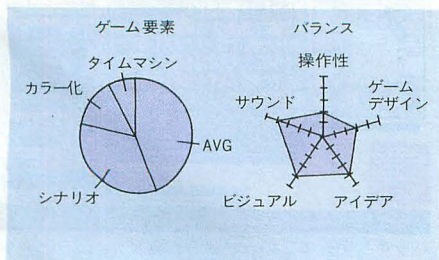
カサブランカに愛を

タイムマシンで時空を飛び越え展開するアドベンチャーです。特にこのX68000版は，総天然色版で楽しめます。

▶私がいままでに解いたアドベンチャーのなかで，印象に残っているのを挙げるとすれば「サザンクロス」（バンダイ），「アルファ」（スクウェア）そしてこの「カサブランカ」であろうか。この作品，2年前にX1 turbo用に発表されたものの移植で，私は前作は残念ながらプレイしていない。今回の作品は，絵はカラーになっていたりして（オリジナルは白黒），X68000の機能をなかなかに使いこなしている。ストーリーはSF小説でいうタイムパラドックスものだが，これが実に巧妙で，よく練られている。2重，3重に時間を越えた事件が絡み合い，エンディングではきっと，涙を誘われるであろう。ゲームは「マワリ　ミル」がキーポイント。BGMは『マイコンBASIC Magazine』やX68000スペースハリアーで有名なYu-You氏のサンプリングとFM音源を駆

使したオリジナルBGM。これはなかなかのものです。誰かBGMモード見つけたら教えてね。

熱中度▶▶▶▶▶▷▷　(善)

▶これは2年前に，X1turbo用に発売された同名ゲームのX68000への移植版です。X68000版はBGMが付き，絵が全面的に描き直されています。白黒だった絵には色が付き（過去の世界では残念ながらセピア色のままだが），新しい絵も何枚か追加されました。素晴しいオープニングも新たに作られています。しかし，これは昔のアドベンチャーです。ストーリーのアイデアは感動的なのですが，作品を構成する世界の狭さはどうすることもできません（2年前は最高水準だと思っていた）。反応してくれる単語の少なさ，鬼のような言葉探しは昔のままです。かつて絶賛されたこのゲームも，2年という歳月のなかではこうも色あせてしまうのでしょうか。昔のアドベンチャーを移植する場合には，どんなに過去の作品のできがよくても，グラフィックやBGMなどの見せかけだけではなく，ゲーム内容やシステム自体にも手を加えてほしいと思います。

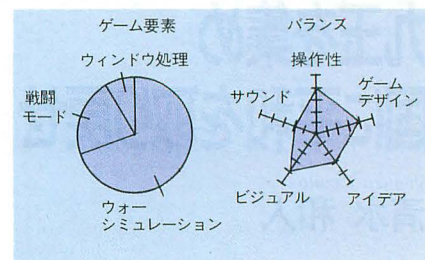
熱中度▶▶▶▷▷▷▷　(A.N.)

X68000用　5"2HD版2枚組　7,800円
シンキングラビット　☎0797(73)3113

SUPER大戦略68K

ようやく大戦略シリーズがX68000にも登場です。迫真のリアルタイムファイトをお楽しみください。

▶Macばりのウィンドウがカラーで現れる。マップをイメージで読み込む。ウィンドウをスクロールさせ，Red Bearのところで指が止まる。プレイヤーを変更する。私はソ連軍を選択し，コンピュータ軍に立ち向かう。CRTに向かい，地形図を右上に小さく配置し，稚内上陸を目指しT-80を生産する。北海道には最新鋭軍を与えてあげる。戦闘が始まる。5つのウィンドウ，ビューマップ，地形図，部隊表，インフォメーション，基本性能表がところ狭しで，東京の住宅密集地のよう。ウィンドウは連動しているから，基本性能表の下敷きになった部隊表では，対象のユニットが反転し，まるで石の下でうごめくアリのよう。不幸にも北海道の最新鋭軍と青森は三沢基地の青い米軍とが衝突し，レオパルドとエイブラムスが画面の上手と下手から登場し，パパパンと戦って一瞬に去る。コーヒーを飲んでいると眠くなったので，コンピュータ同士戦わせて寝る。次の朝，起きるのが怖い。

熱中度▶▶▶▶▶▷▷　(K)

▶大戦略IIかという期待を裏切って，SUPER大戦略68Kが登場した。このSUPER大戦略の他機種のものは，戦闘機でヘリを落とすのは効率が悪い，武装変更ができないなどの弱点があった（F-14で輸送ヘリが落とせないのは悲ピー！）。ところが，この68版ではそれらが改善されていたのだ。これまで大戦略II（98版）とSUPER大戦略（turbo版）をプレイしてきた私にとって，これらの改善はのどのイガイガを取ってくれたかのように思えた。

しかし，ここまで読んですぐに買おうと思った君，君はまだ甘い！　悲しいことに思考ルーチンの強化&時間短縮が，う〜んとうなってしまうのであった。

確かに画面はきれいだし，敵の隣のヘクスに来たからといって自動的に止まってしまうこともない。でも，せっかくX68000用に新しく出してくれたのだから，せめてもう少し工夫が欲しかったなあ。面白いゲームだからこそ，私は次作に期待したい。

熱中度▶▶▶▶▷▷▷　(澤)

X68000用　5"2HD版2枚組　8,800円
システムソフト　☎092(714)6236

がんばれ！リボルティーII

この前，Oh!X編集室にリボルティーIIという，X1turbo用のシューティングゲームが届きました。これはスーパーレイドック以来，X1では久々のシューティングゲームだったのですが，実際にプレイしてみると結構いけるんですよ，このゲームって。

いっちゃなんだけど，私は自分がX68000買ってからというもの，「X1のシューティングってスクロールはガタガタだし，遅いし，ハード的にあんまり未来は明るくないなぁ」，なんて自分勝手に思ってたんですよね。でも，改めてこのリボルティーIIを見せられると，X1もまだまだいけるもんだなー，なんて改めて驚いたりしています。

そういえば，このゲームを作ってくれた「風雅システム」っていうソフトハウスは，X1のゲームというか，シャープのマシンは初参入のはずですよね。それでもって，ここまでのゲームを作っちゃうんだから，ほんとによくやってくれたものだと感激してしまいます。

でも，よくよく考えてみると，これはほかのソフトハウスさんのがんばりが少し足りないせいなのかもしれません。確かにシューティングゲームの開発っていうのは，斬新なゲーム構成を考えるだけでもたいへんだと思うけど，やっぱりシューティングといえばゲームの必修アイテム。うーん，やっぱり作り続けてほしいですよね，いつまでも。　(で)

THE SOFTOUCH

●新九玉伝

九玉を集め国に平和を取り戻せ

Shimizu Kazuto
清水 和人

小坊主コンビが活躍したのは2年前。そしてここに新しいスター「ミンキー涼」を迎え，ビジュアルも一新し再び「九玉伝」が復活した。しかし，どんなにスタイルが変わっても，あの明るさだけはやっぱり変わってないのね。ああ，よかった。

X1turbo用　5"2D版3枚組　8,800円（2ドライブ専用）
テクノソフト　☎0956(33)5555

あれは1987年の8月号，Z'sSTAFF PRO-68Kなどの紹介とともに，このゲームの前作にあたる「九玉伝」のレビューが載っている。それを担当したのは，なにを隠そうこの私なのだが，それだけにこの「新九玉伝」が，前作とあまりに違うのには驚かされた。事実，マニュアルにもそのように書かれてはいるのだが……。

とはいえ，町や村で敵をやっつけながら先に進むという基本的な路線からはハズレてはいない。その代わりストーリーや画面構成がまったく異なるものとなってしまっている。私個人としては，前作の流れを知っている人のためにもう少しサービスをしてくれてもよかったような気もするが，逆に，前作を知らなくても伸び伸びとプレイできるのは結構なことである。

さて物語は，数百年前の昔にあの「ちんねん」と「そんねん」の2人の小坊主が退治したはずの龍鬼が復活し，封じ込められていた九玉が四散してしまったところから，映画のようなアニメーションを使ったオープニングで始まる。今回はなぜか主人公は「ミンキー涼」というミーハーな名前である。そして，ミンキー涼の大冒険は，国王の住むイチダスの町から始まるのであった。

まずはイチダスの町

前作の舞台となったのは，寿村とか椿城下町といった，古風な日本らしい地名の村だったが，今回は九玉が外国に散ってしまったらしい。最初は「イチダス」という町からのスタートである。

のんびりとした音楽が流れるなか，ミンキー涼の足取りは軽い。ゲーム開始直後にすぐ上にある橋を渡って外に出てはいけない。そっちの森や原っぱにはモンスターがうようよいて，ぶつかったら一撃でノックアウトである。まずは王に会って装備を整える。これがRPGのお決まりの儀式である。

しかし，国王はそう簡単によそ者には会ってくれない。そこはそれ，町の人にしつこく聞いて回れば，ほら，最初は誰に会えばいいのかわかったでしょ。

そうしてある人物に会ってみると，この人，最後まで正体のわからない謎の人物ってことになっているけど，額に脂汗流しながら龍鬼によって苦しめられている事情を一生懸命語ってくれるし，おまけに水戸黄門の大事な小道具を貸し出してくれるのだ（風車の弥七はついてこないけど）。

そんなこんなで，玉をひとつだけもらって，次のニラスって町に行けばいいこともわかった。うーん，イチダス，ニラス，サンナスビかひょっとして，これは。とかなんとか思いながら，さっそく旅支度さ。フンフン，これまた取って付けたように武器と防具で王様からもらったお金はピタリとなくなってしまう。

そうして，買い物が終わったら装備することを忘れずにチェックして，さあ，森へとバケ物退治に出かけるのであった。

森では敵を倒すたびに「EXP」が上がる。状態を見るウィンドウを開くと，「NEXT」という項目があって，最初は「100」という数字が表示されている。EXPがこの100までくるとファンファーレが鳴って，レベルアップされる仕組みである。さらに，ある敵はやられたあと青や赤のクリスタルを残していく。これを集めて町や村にある宿屋のオバサンに買ってもらうと，新しい武器や薬を買うことができる。

薬屋で「不死木の実（200G）」や「鬼背木の実（400G）」を買っていれば，即座にHPを回復できるので，用心のために持っているのもいいだろう。「悪絵璃薦水（64800G）」（アクエリアスか，これは？）は，高くてとても買えないが，死んでしまったときに，これを使うと復活できるという便利なもの。ゲームの最後のほうで使える。金が余ってきたら買うか，誰かからもらってしまおう。

お次は2番目，ニラスの町

ニラスの町では，お待ちかねの「チンネーン・ソンネーンブラザース」に会える。この2人が粉々になった九玉の玉をもとに戻してくれるのだが，先にモンスターたちに捕らえられてしまった弟のソンネーンを助け出すために，森のなかへと出かけなければならない。ここは簡単な迷路になっているが，まず入ってすぐのあたりにパワーリングというものが落ちている。これはあとで必要だから，必ず捜し出そう。

さて，この迷路の先には行き止まりの壁がある。どうやらこの壁の向こう側に行かなければならないのだが，そのためには村人から話を聞いていないと，どうにもなら

ハッキリいってこの人が謎の人物です

ない。

そうしてなかに住むモンスターと大格闘して勝つと，ソンネーンに会える。さあ，これでチンネーンの家に行けば，最初の玉「カウンボール」が手に入る。この玉の効用は解毒，このあとすぐに必要になる。

ドラーク沼という毒草の繁った沼を通って，目指すは次の「サンタピアノ」の町。なぁんだ，サンナスビじゃなかったね。

2つの玉を手に入れろ

サンタピアノの町あたりから，そろそろ忙しくなってくる。まずは，複雑な迷路を攻略しなければならないから，そのための装備を整えることを忘れてはいけない。

この町では，町長の娘が誘拐されてしまったらしい。彼女を救出するために，盗賊の館に潜入するのだ。この館の内部が複雑で，通り抜けたら方向感覚を養っておかないと必ず迷子になってしまう。ここでもボスキャラをやっつけることになる。

とにかく，このゲームに登場するボスキャラは，体の一部に必ず弱点を持っているから，戦いの最中に捜し出して効率的に攻めれば，比較的簡単に倒せるようなる。

また，この町のボスキャラに出会うためには，重い扉を開けるためのアイテムが必要になる。実際のボスキャラとの勝負はEXP次第だが，これもサンタピアノの町に来る途中のドラーク沼のボスキャラにダメージを受けないようなレベルを維持していれば，まず大丈夫。意外とあっさりと倒すことができる。そうしてクリアすると，敵を攻撃する力を持った「アームボール」と「イクスボール」が手に入る。やったね。

デッカイ迷路の4番目の町

シリューダの町（ヨンなんとかではなかった）では，大きな迷路がある。ここではまったくヤンなっちゃうほどマップを作るのはたいへんなのだが，いざ完成して全部の道がつながってみると，「やったね」という気分になった。久々のこのテの作業にすっかり満足。

この迷路は，へんな小屋から入る廃坑なのだが，なかには離れた場所から敵をやっつけられるボールと，ヘンな名前のアイテムが1個隠されている。どちらも結構，奥深い場所に隠されているのだが，これがなくては先に進めないので，命がけで取りに行くしかないのだ。それには急がば回れでマップを書きながら進むのがベストのようである。

シリューダの町には，このほかにもサンタピアノで手に入れたあるものとボールを交換してくれる人物がいる。この人に会うのがここのポイント。あとは，ここの町の人たちの話はよく聞いておかなければならない。この先出かける湖には秘密が隠されていて，話を聞いていないことには，どうしていいのかわからなくなってしまう。

人の話は真面目に聞かなきゃ先に進めない

弱点さえわかればあとは比較的ラク

終わりは近いぞゴマンスの町

この町には迷路こそないが，いよいよ終わりが見えてきたような気がする。とにかくさまざまな情報が隠されているのだ，この町には。やはり町の人の話が大事になってくる。なんかこう書いていくと，簡単なRPGのように思えてくるが，実際，2，3カ所のヤマ場さえクリアすれば，あとは比較的口当たりの軟らかいゲームなのである。不条理なストーリーは一切なく，「リーズナブルな雰囲気で気楽に遊んでもらおう」という制作意図には好感が持てる。

さて，このゴマンスの町ではとてつもなく大きな買い物をしなければならないので，貧乏人は先に進めない。せっせと敵を倒してクリスタルを集めておこう。赤いクリスタルは高く売れるので，これを持っている敵を集中的に攻撃するといい。また，売る場合はできるだけ遠くの町のほうが，相場が高くなっていて儲かる。それと宿屋では，クリスタルを買ってくれるほかに，宿泊するとHP，MPの両方を回復してくれるので，たまには休憩するのもいいだろう。

残すは2つの町だけよ

5の次は「ロッテス」の町だ。ここでは強力な武器が手に入る。これは毎度お馴染み迷路のなかに隠されている。今度の迷路はなんと酒場の隣から入ることになっている。少々このあたりの展開は無理があるようだが，迷路そのものはわりと簡単なので気軽にフラリと入ってみるのがいい。

それからこの町には雑貨屋という便利な店があるので，品定めしてから買い物をしよう。いいかげんな物を買ってしまうとあとで困ることになる。また，この店に入ったとき，条件を満たしていないとなにも買えないので，確認するためにも必ず寄ってみることだ。私の場合はボールが1個足りなかったみたいで，悩んでしまった。

それにここの宿屋は，クリスタルの買い取り相場が高いので非常にラッキーである。

それから，いざ島へモンスター退治へと出かけることになる。するとそこでまたボールが1個手に入る。

お次が最後の町「ラースト」である。てっきり，「ナナなんちゃら」という町だと思っていたけど，この名前のほうがわかりやすくていいや。しかし，最後の町にふさわしいはずのここには，店も宿屋もない。あるのは長老の家とばあさんの家だけだ。

町はずれの森に出かけてみると，あれを持っていないとダメ，これもなければダメと，いろいろと不足しているとあれこれ悩んでしまう場所だ。しかし，ここで九玉はすべて揃えられる。

そうして××を××して，××に持って行って，そのあとこの××が××して……，とにかくこの××が，このあといっぱい続くと思ってくれれば間違いない。これを無事クリアすればエンディングね。さて，最後のキーマンはいったい誰か？　そこに待ち受けるものはいったい……，大いに好奇心をそそっておいて，このレビューは終わってしまうのさ。

＊　＊

時間の経つのをすっかり忘れて，プレイしてたこの私も，エンディングの画面と音楽に，しばしの幸福感を覚えていた。「最後の戦いはちょっとキツかったな」。そうつぶやくと，データのいっぱいつまったディスクをそっとX1から抜き出した。

ようやくこれで，昨年末から続いていた戦いが終わったかのように思えたのだが……，えー，このあとボスコニアンにM&MIIと続くの？

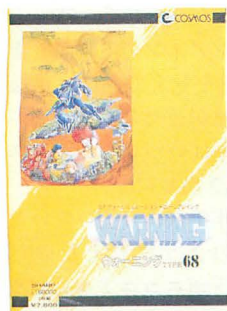
やれやれ，年号が変わっても戦士には休息はないのね，やっぱし。

THE SOFTOUCH

●ウォーニング

迫る海賊追い払い 目指せ宇宙の大商人

Ogikubo Kei
荻窪　圭

宇宙で貿易をしてお金を儲け，迫り来る海賊たちを蹴散らしながら行方不明の兄を捜し出すという，この「ウォーニング」。金儲けがヘタだとなにもできないという，なんともシビアなウォーシミュレーションなのです。

X68000用　5″2HD版2枚組　7,800円
コスモス・コンピューター　☎03(770)1821

このウォーニングには，いくつかのドヒャーがある。少なくとも，私にはドヒャーだったのである。だって，誰がどう見ても，ただのSF（風）RPGじゃないか。アニキである「ビリー」を捜し出すという能書きがあろうとも，RPG以外には見えないではないか。そして，RPGではないのかというと，絶対にシミュレーションなんかじゃなくて，RPGなのである。変わったRPGである。

まず，ドヒャーその1は，なんと，「経験値」がない！　のドヒャーである。まあとりあえずは，適当にやっていれば適当に強くなって，先のことはそれから考えよう，といった脳天気RPGだと信じていた私はいきなりこれでは話が違う，ドヒャーなのだ。経験値がないということはどういうことかというと，「戦ってもこっちが消耗するだけで，力の無駄」ということである。しかし，敵さんは容赦してくれないのである。「ごめん，いま戦ってる暇がないんだ」とでもいおうものなら，怖い顔をして大事な積荷を奪っていってしまうのである。

ドヒャーその2は「職業が行商人だ」，ということである。英雄でもなんでもなくて，人望があるわけでもなく，なんの伝説も背負わない行商人なのである。「安く仕入れて高く売れ」，それでなきゃ生きていけまへんの商人根性が必要なのである。物を売って金にするのは大昔のRPGから基本ではあるが，「物を売らなきゃ一切金にならない」RPGは初めてである。ドヒャーである。

第3のドヒャーは，「高けりゃいいってもんじゃないのさ」である。たいていのRPGは，「金がたまれば高価な装備」が基本だった。しかし，高い武器がいいとは限らないのである。すべての武装について，長所と欠点があるのであって，高いからきっと役に立つだろう，などという甘い考えだと，ただの金の使い方を知らない成金である。たとえ5,000円の機動兵器が買えたとしても，やはり私は1,200円で買ったボフォースを愛用してしまうのである。

第4のドヒャーは，「なにひとつとして成長しない」である。最近の，使い込めば使い込むほど強くなるような，都合のいいアイテムはどこにもないのである。成長したければ，プレイヤーが頭を磨き，金にものをいわせて最適の装備をひねり出すしかないのである。

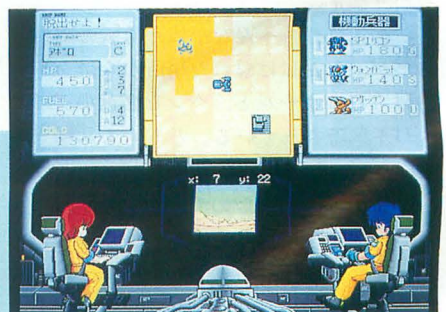
ドヒャークラスの特徴はこんなものである。その次に，「ちょっとした心配りも味の内」が付いてくる。たとえば，常に現在地の座標が示されているのである。すると，マップなど作らなくとも，座標さえメモしておけば，どこにだって，いろいろ障害はあるにしろ，行けるのである。それから，宇宙空間ではレーダーが使えるので，適当に船出しても，燃料さえあれば，道に迷うことはない。

あなたはどれに乗りますか？

で，まず，船の名前を入力して自らのアイデンティティを示すところから始まる。私の船の名は，「脱出せよ！」号である。いついかなるときも，シッポを巻いて逃げようとも，生き延びることを目標としている。もし，名前に文字数制限がなかったら，「荻窪圭は金の亡者になった」号なんかがよかったのだが，まあ仕方がない。

で，初級行商人（トレーダーというらしいけど）の駆け出しが乗れるクラスの船は3種類しかない。

私はとにかく丈夫なコンボイのような，日産のサファリとかトヨタのハイラックスのような丈夫なやつにした。これだと砲塔は付けられないが，コンテナがいっぱい積めるので，金は儲かる。私は本来，機動性に優れたタイプのやつが好きなのだが，私がそんなのを操った日には，商売を忘れて，さっさとあちこちの惑星目指して冒険の旅に出かけてしまうのがオチである。だから控えめなやつにしたのだ。この謙虚さが死の淵から救ってくれるのである。

で，プシェード星系（例によって，SFものにありがちなわけのわからない名前）で一番栄えている星に出現する。右も左もわからぬ土地でまずは商売を始めるのである。資本金は何万円かあるので（ホントの単位はゴールドだけど，円のほうが親しみやすい），武器を装備するなり，商品を仕入れるなり，することは山ほどある。きっと，金は足りないだろうから，ショップで1万円借りることになるだろう。ちなみに，初級行商人，中級行商人，上級行商人とあり，このゲーム全体において，ランクづけはこれだけである。しかも，クラスが上がると，どういうメリットがあるかというと，でか

惑星上には放射能事故現場もある

くていい船を買えるようになるだけという免許みたいなもので，ただそれだけだが，これは重要である。

行商人はどん欲である

仕入れた品物は，ほかの星へ行って売る。星間貿易というやつである。関税もなにもないから，純粋に買値と売値だけ気をつけていればいい。あと，それと，運送費ね。きょうび，ガソリン代もバカにならないし，いつ海賊に襲われるかわからないから武装にも費用がかかる。オマケに襲われて，戦って傷つけば，修理費もかかるし，てなぐあいで，なかなか難しい。「行けぇ！　紀国屋文佐衛門，嵐の海へ船を出せ！」といった世界でもあるのだ。

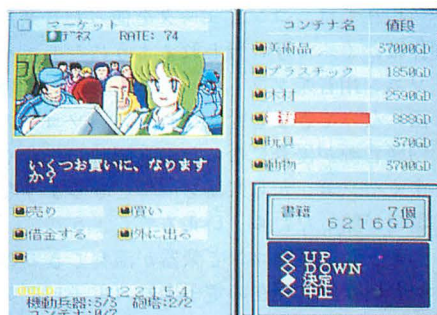
右も左もわからない初級行商人は，コンピュータバンクへ行くのがいい。そこで，いろいろな惑星の話を（有料だけど）聞かせてくれる。たとえば，アルテモは農業が盛んだ，とか，デネスは商業の発達した首都的役割の星，だとか，クレーゴは工業の星だ，とか，モントには鉱物資源が豊富などなどである。よく考えてみれば，惑星というのは，恒星の回りを公転しているもののはずだが，そんな恒星はどこにもない。不条理な宇宙だ。

最初に出現するデネスに最も近いのがアルテモだから，まずは，アルテモ〜デネス間で，穀物と衣類の貿易でも営むのがよいであろう。星から星へと大気圏突入をすると，いきなり見知らぬ荒野に着地して困ってしまうが，着地地点は常にその星のスペースポートのやや南だから，すぐに人里にたどり着ける。

ある程度儲かって，武装も揃ってきたところで，足を伸ばしてクレーゴへ行くと，機械やトラクタを売っているという寸法だ。

しかし，である。相場，というものが星ごとにあって，しかも変動するので，「この相場なら儲かったのに，グヤシー」という悲劇はあとを絶たない。そのうえ，星によってガソリン代も異なるので，これも考慮に入れておかないと，利益が少なくて運送費を入れたら足が出た，ということにもなりかねない。一番ガソリンの安いのが，ウロボリスに近い腐食ガスに覆われた不毛のブリックである。とにかく，最初は地味に，小金をため込むことが必要だ。

でも，冒険をしたい年頃であった

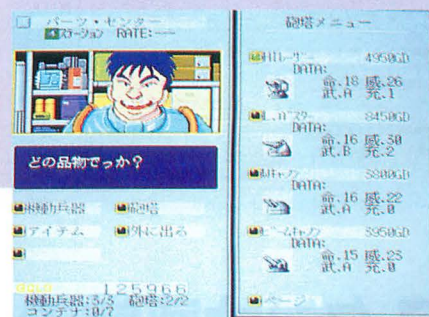
しかし，商売だけをしてのほほんと暮らしていくには，この星系は意図がありすぎ，謎がありすぎ，複雑すぎた。アステロイド

ここが貿易の拠点になるマーケット

ベルトに囲まれた，どうしても行けそうにないステーション。果てのある，宇宙。きっと，その向こうでは，宇宙が滝のようにドドドドドドッと4次元にでも注いでいるのだろう。そして，複雑な宇宙のなかで宇宙船は思ったよりも後続距離が短い。冒険などを始めると，あっという間に，ガス欠になり，救助信号の世話になる。

自由に行ける惑星は7つあるが，全部を回るだけでも大騒ぎである。ウロボリスには近づかないようにしないといけないし（帝国軍がうろつくのは全マップの4分の1近くである），海賊やローグなんかもうろうろしているし，なんといっても，商売をしながら旅をしないと，あっという間に金は底をつく。かといって，ただ物を運べばいいというわけではなくて，売れ筋を運ばねばならない。私ゃか弱い富山の薬売り。次に行く星を考えてルートを決め，品物を仕入れないと，商品がダブついたり，引き取ってもらえなくて積み荷が増える一方だったり，と，ロクなことがない。特に，星間の距離が長いクレーゴ（あるいはモント）〜フリード間では，積み荷の選択ミスが貧乏を招く。

しかも，ただ貿易しているだけでは人生が退屈なのである。見知らぬ惑星に着いたら，探検してみたくなるのが人情ではないか。すると，クレーゴやフリードでチェルノブイリ爆発後の放射能区域や（私はそこを調査したかったのに，オペレータの女の子が危険だからだめだという），アルテモで遺跡や，モントで温泉などを見つけられたりする。しかし，探検ばかりしていると，途中で出会った怪物やバンデッドとの戦いで，修理費がかかってよろしくない。

私は，じっくり腰を据えて，一歩一歩進んで行くことに神経が耐えられない類のどうしようもない人間である。つい，冒険をしてしまう。装備が万全でないのに，つい，遠出をしてしまって，救助されたり，売れない在庫を抱えて途方にくれたり。このRPGの主人公は明らかにプレイヤーのもの

うっかりと連邦軍にケンカを売ってしまう私

怪しげな店員のいるパーツセンター

でありすぎ，ああ，結局，名古屋人の私はがめつい大阪商人にもなれないのだ。おかげで，この星系のことは隅々まで詳しいけれども，それが生かせるだけの資本力と，宇宙貿易組合が認めてくれるランクが足りないので，いつも苦汁をなめている。

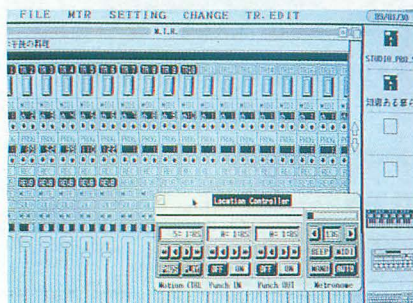
地道なことが嫌いだから，武装も一気にドッカーンとやってしまう，ミサイルランチャーをつい買ってしまう。レーザーで敵をひとつずつつぶしていくほうがよいかもしれないのに。そのうえ，戦いが嫌いだから，つい海賊だろうが，ローグだろうが，タコのお化けだろうが，話しかけてしまい戦いが後手後手に回ってしまう。

発展的解消

と，いうわけで，結局はマネジメントRPGであるウォーニングは，とてもユニークなゲームなのであるが，まだまだ開発の余地があるといわざるを得ない。プレイヤーを引っ張って行くにはいささかイベントが足りないし，じっくりハマリ込むには味付けがサッパリしすぎている。

現状のゲームシステムでも，表現方法とバランスに工夫を加えれば，かなり見違えたものになるだろう，とも思う。宇宙空間で貿易をして，なおかつ戦闘も楽しめるRPGといういいコンセプトを持っているのだから，もし続編が出るのであれば，ぜひ見てみたい気がする。

とにかく，ウォーニングの世界は，アニキを見つけたあとでも，きっと延々と続くのである。貿易商人の，明日はどっちだ。

●Musicstudio PRO-68K

スタジオ感覚でレコーディング

Shiina Kaoru
椎　薫

ついに発売されたX68000用MIDIボード。人気が高く，ちまたではMIDIボードの品切れ続出とか。ここでは，MIDIソフト第1弾，Musicstudio PRO68-Kの機能と操作性を検証してみましょう。X68000のミュージックシーンは，いま始まります。

X68000用　5″2HD版2枚組　25,800円
シャープ　☎03(260)1161

現在では，コンピュータを使った音楽制作はごくふつうに行われるようになり，パソコン用音楽ソフトのバリエーションも豊富になってきました。これも，パソコンの内蔵音源が優秀になったのと同時に，たやすくパソコンから楽器を演奏できるような環境が整ったためといえるでしょう。

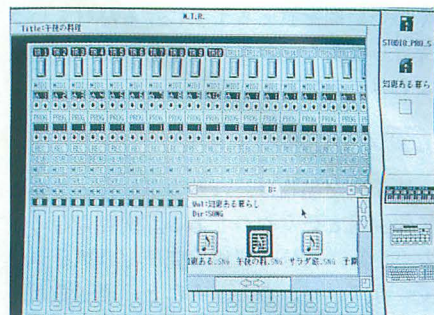
どうしてパソコンで楽器が演奏できるのかというと，現在の電子楽器（電子ピアノ，シンセサイザなど）のほとんどが，MIDI（ミディ）と呼ばれる共通信号によって外部コントロールできるように設計されているからです。MIDIとは演奏情報を楽器間で自由に通信し，互いにコミュニケーションできるようにと決められた規格です。国内外の主だった楽器メーカーによって決められた規格のため，メーカーや機種は問いません。パソコンでMIDI信号を作って楽器に送れば，MIDI対応の楽器であればなんでも演奏できるというわけです。

ところで，音楽専用に設計された特別なパソコン以外，MIDIを出力するような機能は持っていません。もちろんX68000にもそんな機能はありませんね。そこで必要になるのが，MIDIインタフェイスボードです。パソコンはMIDIインタフェイスボードを取り付けることにより，初めてMIDI信号を扱うことができるようになるのです。

皆さんもご承知でしょうが，このたび，X68000にもMIDIボードが発売されました。X68000でも，ようやく本格的なコンピュータ音楽が可能になったわけです。そして，そのMIDIボード用ソフトの第1弾として発売されたのがMusicstudio PRO-68Kです。これはリアルタイムマルチ録音を主体とする音楽演奏用シーケンサソフトです。

まずはセットアップ

Musicstudio PRO-68Kを起動させるには，まずMIDIボードCZ-6BM1をX68000に装着しなければなりません。MIDIボードは，出荷時の初期設定（割り込みレベル4，MIDI OUT ON）の状態でそのまま起動しますので，そのまま取り付けてください。

X68000でMIDIを使用した音楽制作を行うには，MIDIボードとMusicstudio PRO-68Kのほかに，当然ながらなんらかのMIDI楽器が必要です。MIDI楽器の後ろを見ると，“MIDI IN”/“MIDI OUT”と書かれた5ピン端子を見つけることができます。これがMIDI信号を入出力するための端子で，MIDI端子と呼ばれています。これがMIDI楽器の証です。MIDIボードにはスペースの都合からか，ひとまわり小さい端子がついており，専用の変換ケーブルでふつうのMIDIケーブルに接続するようになっています。

24トラックのMTR

楽器の接続はいたって簡単，MIDIボードのMIDI OUT端子と楽器のMIDI IN端子をMIDIケーブルでつなげばOK。これで演奏情報を楽器に送信できます。ところで，CZ-6BM1には2つのMIDI OUT端子がありますね。ソフトによっては独立制御できますが，とりあえずどちらに接続しておいてもかまいません。

単に演奏を楽しむだけなら，この接続だけでかまいませんが，Musicstudio PRO-68Kには，楽器からの演奏情報を録音する機能があります。楽器からの演奏情報をX68000に取り込むためには，さらに楽器のMIDI OUT端子とMIDIボードのMIDI IN端子を接続する必要があります（これは，後述するリアルタイム入力で必要）。

ところで，MIDI端子とはいっても，決して特殊な端子ではありません。これは汎用のDIN規格（JIS規格のドイツ版）で定められた端子で，MIDI以外にもカセットデッキなどのオーディオにも広く使われています。したがってケーブルもわざわざ専用のものを買う必要はありませんが，MIDI専用となっているケーブルは5ピンのうち3ピン（うち1ピンはGND）しか使用しておらず，ほかの端子を接続していないケーブルも多々あります。オーディオ用に使いまわす場合には注意してください。

なにができるのかな？

レコード制作では通常，マルチトラックレコーディングという手法が使われます。これは，ベース，ドラム，キーボード，ギター，ボーカル……といった各演奏パートを1つひとつ別々に録音してゆき，全パートを入れ終わったら，バランスを取ってステレオでうまく聞こえるようにします（これをミックスダウンと呼びます）。ここで使われるテープレコーダは，24トラック以上です。ちなみにラジカセなどは4トラック

ですが，２トラックずつステレオでＡ面，Ｂ面として使われています。

ラジカセなどでは，ステレオ録音として２トラックずつ同時にしか録音できませんが，マルチトラックレコーダは，各トラックごとに自由な録音/再生ができるようになっています。カセットタイプのものにも４トラックマルチレコーダなどが数多くあり，マルチレコーダも自宅での多重録音に広く利用されています。

実は，こういったマルチトラックレコーディングをパソコンだけでやってしまおうというソフトがMusicstudio PRO-68Kなのです。ただし，マルチ録音するのはあくまでもMIDI信号です。よく間違えられるんですが，MIDI信号は音そのものではありません。MIDI規格に従ってデジタル符号化された演奏情報であることをもう一度確認しておきましょう。

したがって，録音したデータをプレイバックする場合，接続する楽器を変えれば音も変わってしまうわけです。逆に，同じデータでもあとから自由に音色を変えたりできます。このあたりがふつうのテープレコーダとは根本的に違うところです。デジタルの最大のメリットは，なんといっても，あとから自由にデータを変更できる点にあるのです。

機能を見わたすと

Musicstudio PRO-68Kを立ちあげて，用意されているメニューを開いてみてください。まずは，ビジュアルなシェルに驚かされます。マウス操作はX68000本体付属のビジュアルシェルに準拠して作られていますから，とまどうこともないでしょう。

Musicstudio PRO-68Kのメイン画面からメニューバーの“MTR”をオープンすると，まさにスタジオのミキサーというMTRウィンドウ画面が登場します。画面は，ウィンドウの奥に展開されていて，ウィンドウで自由な場所が覗けるような仕組みになっています。もちろんウィンドウのサイズや位置は自由自在です。

ミキサーは24トラックまであり，ウィンドウを最大にしても全部は表示しきれません。そこで画面をスクロールさせることで全部を見わたすようになっています。

また，カーソルを右端に持っていき，どんどんスクロールさせていくと，MT-32のDrumコントロール画面が現れます。ここではMT-32の各ドラムパートの音量バランスや，ステレオOUT時の定位を決めることができます。各パートのネームに当たる部分をクリックすれば，それぞれの音が鳴りますから，接続が正しくなされているかのチェックにもなります。

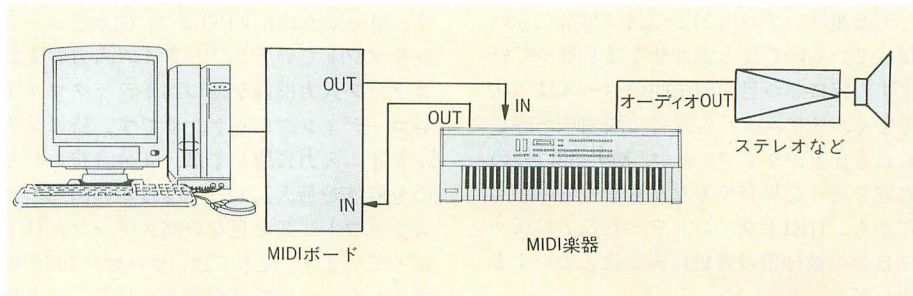
これらはMT-32専用の機能ですが，MT-32以外でもドラム音源を内蔵した楽器やドラムマシンを接続して同様にドラム演奏をすることも可能です。要は各ドラムパートに割り当てられているノートナンバーとMT-32に割り当てられているノートナンバーをうまく一致させればよいのです（詳しくは各楽器の取扱説明書を参照）。

MT-32のDrumコントロール

ところで，画面にはたくさんのコントロールスイッチが用意されており，なにやら難しそうですが，同じモジュールが24個並んでいるだけと思えば簡単，操作に臆することは決してありません。まずは，ひとつのトラックモジュールがひとつの楽器を担当していると思ってください。

デモ曲を聴く

ディスクに用意されているデモ曲の演奏

図１　MIDIシステムの接続

MusicstudioとMT-32

MIDI楽器ならばどんな楽器でも接続して演奏できますが，同じMIDI楽器とはいえど，メーカーや楽器の種類によってその機能はまるで違います。特にパソコンに接続する場合のポイントとなるのが，マルチ音源であるかどうかという点です。

基本的な音源は１楽器１音色の発音と考えてください。ピアノ曲のようなものならば１台でよいのですが，オーケストラのような演奏をするには，いろいろな音を同時に出す必要がありますね。もちろん演奏パートの分だけ楽器を用意できればいいのですが，これではたいへんですね。そこでマルチ音源と呼ばれる音源を利用することになります。

マルチ音源とは８音色程度の同時発音ができ，各音色を独立して演奏できるような音源をいいます。つまりマルチ音源は，１台の中に複数台のキーボードを詰め込んだものと考えられ，まさにコンピュータ演奏のための音源といえます。ただし，機能が多い分だけ操作も複雑になるのは避けられません。多くのマルチ音源は，初心者にはかなりコントロールが難しい仕様になっています。

そこで，おすすめなのがローランドのMT-32というマルチ音源です。Musicstudio PRO-68Kでも，このMT-32という音源を基準にしてサンプルのデモンストレーション曲が作られていたり，基本セッティングがされています。

MT-32は，数あるMIDI音源の中でもコストパフォーマンスに富んでおり，扱いも簡単なのでパソコンミュージックの入門用としては最適の楽器といえます。MT-32はコンパクトなボディにもかかわらず，最大８音色の独立演奏ができるほか，ドラム音源をも別に持っています。そしてリバーブを内蔵しているなど，１台でもかなりハイレベルな演奏が可能です。見かけよりも高度な使い方ができるようになっていますから，初心者向けといっても決して侮れません。

今後，MT-32に対応したゲームソフトなども予定されているようです。

ただし，MT-32は純然たる音源ですから，鍵盤などはついていません。キーボードを弾いてそれを録音するというのがMusicstudio PRO-68Kの得意とするところですから，MT-32となんらかの鍵盤つきMIDI楽器を揃えておくのがもっとも理想的です。しかし，リアルタイム入力などには手を出さない方はとりあえずMT-32を購入されるのがよいでしょう。

なお，なにはともあれ使ってみようという場合は，MT-32をリセットして初期状態にしてから使用するとわかりやすいでしょう。リセットは，MT-32のマスターボリュームボタンを押しながら，リズムボタンを押し，“ALL Reset OK”と表示されたらPART１ボタンを押してください。

MT-32

をしてみると，MTR画面の動きが確かめられます。ファイルウィンドウで，デモデータをロードしてみてください。ウィンドウのオープンの仕方や，ロード中に出てくる砂時計など，ちょっとビジュアルを意識しすぎかなとも思われますが，これくらいの余裕がなければ遊びもできません。画面の色を自由に変えられるコントロールパネルやスクリーンモードの切り換えなど，その遊び方の凝りようにも感心するソフトです。

ロード後にロケーションコントローラのウィンドウをオープンして，“PLAY”をクリックすればデモ演奏がスタートします。すると演奏に合わせてMTR画面のボリュームやレベルメータなどがシンクロして動き出します。その動きのスムーズさは，なかなかのものです。

デモ曲はいずれもMT-32を120%ぐらい使っているのではと思わせるほど凝っています。特に2つ目のデモ曲のベースはノリもよく，リアルタイム録音の醍醐味が感じられる仕上がりです。また，MT-32以外の音源を使った場合でもなんらかの音は出ますから，HELPウィンドウの音色名に従って自分の機種用の音色に置き換えていくとよいでしょう。

演奏をしながらでもマウスによるスイッチのコントロールが可能なため，演奏状態をリアルタイムで変化させて楽しむこともできます。たとえば，ボリュームスライダーを動かせば，該当するパートの音量が変わり，プログラムを変えれば音色が変わります。

また，各パートがそれぞれどんな演奏をしているかを知りたければ“SOLO”を押します。すると，該当するパートのみの演奏を聴くことができますから，演奏の仕組みを知るうえでとても参考になります。

そのほかにも，リバーブ，PAN(定位)，MUTE(演奏しない)，MEMO(メモ書き)，MIDIチャンネル，TUNE(微妙に音をずらす)，TRNS(移調)などが，各モジュールごとに設定できます（ただし，リバーブやディチューンなどはMT-32専用の機能ですので注意)。

ソフトウェアキーボードを呼び出す

さらには，リアルタイムエディット機能により，マウスによる操作を録音して演奏データに加えていくこともできますから，コンピュミックス（自動ミキシング）さながらの高度な演奏も楽しめます。すなわち，ダイナミックに音量を変化させたい場合などには，2つあるMIDI OUTの片方をボードのMIDI INに接続し，演奏と同時にマウスでスライドボリュームを操作することで，現在演奏中のデータをリアルタイムに更新していくことができるのです。

充実のリアルタイム入力

それでは，Musicstudio PRO-68K本来のシーケンサとしての機能を紹介しましょう。Musicstudio PROには，従来のシーケンサソフトでいうリアルタイム入力およびステップ入力機能を持つ24トラックマルチレコーディングシーケンサです。特にリアルタイム入力に関しては，4分音符あたりの分解能が最大240という，ほかの国産パソコンソフトに類を見ない高スペックに仕上がっています。従来では，せいぜい192が限度でした。この数値が大きいほど，より細かな表現まで録音できるようになります。

また，PC-9801用などで発売されているものに対し，24トラックとチャンネル数が多いというのも特徴です。同一のMIDIチャンネルに，音源が対応していれば複数のトラックを割り当てることができるほか，エディット時には2つのチャンネルで音を比較したり，データを一時保存したりといった操作が多くなるため，チャンネル数で作業効率が格段に変わってくるのです。

さて，リアルタイム入力とはMIDI出力できるキーボードなどの楽器からの演奏情報を直接録音する方法です。テープレコーダ同様，生のニュアンスが再現できるため，楽器のできる人にとっては効果抜群です。

リアルタイム入力するには，まずMIDIキーボードなどが正しく接続されていなければなりません（楽器のMIDI OUTとMIDIボードのMIDI IN)。ロケーションコントローラーのメトロノーム部にあるテンポを指定し，BEEPをONしておけば録音時にガイドクリックが出，MIDIをクリックしておけば，メトロノームをMIDIの音符演奏に置き換えて出力します。テンポはどんなに遅くしても，実際に演奏するときに戻せばOK。難しいフレーズはテンポを落として録音できます。

次に録音トラックを指定します。指定は録音したいトラックにあるモジュールの“REC”をクリックして選択します。また，レコーディングフィルタがあり，不必要なMIDIメッセージをカットすることができます。アフタータッチ（鍵盤を押したあとのキーにかかる力の変化）など，無意識のうちに出力して思わぬメモリを消費するケースや，エフェクトと音階を別々に録音したい場合などには欠かせない機能です。

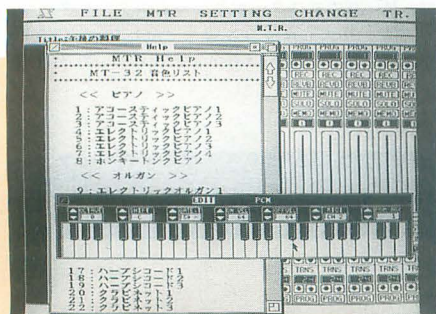
そして“PLAY”をクリックすれば録音がスタートします。ロケーションコントローラの機能を併用すれば，自由な位置から録音したり，部分的な差し換え（パンチイン/パンチアウト)ができるなど，豊富なレコーディング機能が備わっています。

音長補正はクォンタイズで

また，楽器が下手な人でもクォンタイズ（丸め込み）を有効に使えば素早い入力ができます。クォンタイズとは，その部分で演奏されている最小の音符を指定することで，自動的に音長のずれを補正する機能です。リアルタイム入力では，クォンタイズは欠かせない機能といえるでしょう。クォンタイズは，64分音符から全音符まで用意されており，しかも小節の範囲を指定することもできます。どんなに上手でも人間が弾く以上ずれを生じますから，機械的なノリを出す場合にも有効ですし，逆に人間的なノリを残したいときにはクォンタイズしないように指定できるのです。

リアルタイム入力で，もうひとつ便利な機能に16のMIDIチャンネル情報を同時に録音し，トラック分けできることがあります。これは，ほかのシーケンサで作成されたデータをチャンネル別のままMusicstudio PRO-68Kに入力したい場合などに使います。それには，MIDI I/FメニューでMIDI ClockをONにし，外部からのMIDIクロックに同期してシーケンサが動くようにします。外部同期しながらの録音は，シーケンサ機能の中でも特に高度な機能です。そして，録音したいチャンネルを録音したいトラックに指定するとともに，“REC”をクリックしてスタンバイします。

ステップ入力機能

続いて，ステップ入力を利用すれば，リアルタイム入力したデータを修正したり，楽器ができなくても1音ずつ順番に演奏情報を入力していくことができます。なお，Musicstudio PRO-68Kでは，それぞれに有効なステップ入力方法が用意されており，データの修正には，MTR画面の各トラッ

クモジュールにある“ED1～ED24”をクリックすれば，エディットウィンドウがオープンします。入力されているデータが，数値で表示されますから，修正したいパラメータをクリック選択してデータを変更します。

また，新規にステップ入力したい場合は，ソフトミュージックキーボードとステップレコーディングウィンドウを使います。ところでステップ入力では，音符の長さはステップ数で表現されます。そして，ステップ数は分解能によって異なってくるのです。たとえば4分音符あたり240ならば4分音符＝240，全音符＝960です。

あまりステップ数に大きな数字を使うとステップ入力では，わかりづらくなります。ステップ入力のみならば4分音符あたり48でもたいていの曲を表現でき，わかりやすくなります。ステップ数は，細かければよいというものでもないのです。

注意点

ところで，マルチレコーディングタイプのシーケンサにありがちな問題に，ついつい和音データを入れすぎてしまう場合があります。電子楽器には必ず同時発音数というのがあり，それ以上の和音は演奏されません。また，リリースの長い音色を選ぶと，音が重なりあうことが多くなり，発音数が足りなくなる場合もあります。音が不自然に途切れるようになったら，同時発音数を超えている証拠です。

ちなみにMT-32の場合に注意しておきたいのは，MT-32では32パーシャル(MT-32における音を作る場合の基本単位）までの同時発音ができるのですが，これは単純に32音に一致しません。音色の選び方で，発音数が変わってくるのです。というのも，1パーシャルでできた音色もあれば，4パーシャルでできた音色もあるからです。

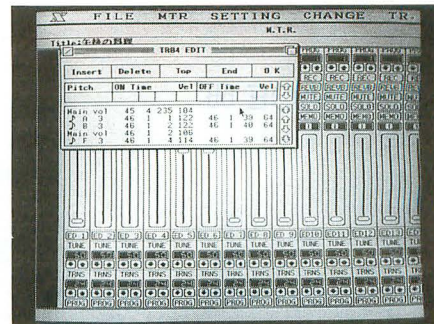
4パーシャルの音ばかりを選べば8音，これでは1パート1音にも満たなくなりますね。Musicstudio PRO-68Kの音源としては，かなりもの足りない気もします。そこで，もう何台かMT-32がある場合は，オーバーフローアサインという特別な機能を使うことで，同時発音パーシャルを拡張することもできます。

まだまだ盛りだくさん

そのほかにも，トラックごとにエディットできるさまざまな機能が用意されています。まず，チェンジノートは，指定小節中の指定ノートナンバーを指定ノートナンバーに変える機能です。これは，ドラムパートで鳴らす音を変えたい場合に便利です。

ドラム音源の場合，ドラムの各イベントは，ノートナンバーが割り当てられ，違う音色を同一チャンネルで送ることができるように作られています。ただし，ドラム音源によってノートナンバーとイベントとの関係は違います。通常ドラム音源には，ノートナンバーの変更機能がありますから，MT-32にあわせるか，それともチェンジノートでMusicstudio PRO-68Kのデータを変更するかしなければならないのです。

チェンジプログラムは音色切り換えを，チェンジリバーブはリバーブのオン/オフ，チェンジパンポットは定位を，チェンジディチューンはディチューンデータをそれぞれ小節の頭に設定する機能です。また，チェンジボリュームは，ある小節からある小節まで連続的にボリューム変化をつけるための機能で，クレッシェンドなどをつけることができます。

このように，PROというだけのことはあるマルチトラックレコーディング機能と性能を持っており，さらに高機能なMIDI音源を組み合わせることでプロにも応えられるMIDIシステムを組むこともできそうです。

トラックごとにエディット

今後，楽譜入力のできるMUSIC PRO-68K〔MIDI〕の発売も予定されていますが，現状でもMUSIC PRO-68Kを使って楽譜入力を行い，データを流用できますので楽器はまるでダメという人でも手軽にコンピュータミュージックを楽しめるでしょう。

MIDIチャンネルとは

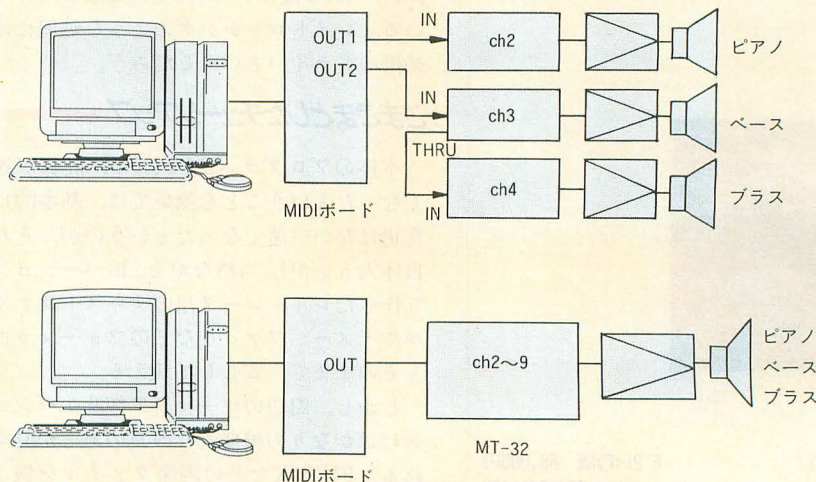
Musicstudio PRO-68Kのような，マルチトラックのシーケンサを扱う上でぜひとも知っておかなければならないのが，MIDIチャンネルの概念です。MTR画面にもチャンネル指定がありますが，これをむやみに変えると思ったとおりのパート演奏にはなりません。

MIDI信号は，1本のケーブルを接続するだけでさまざまな演奏情報を送ることができるという大きなメリットがあります。ところが，接続したすべての楽器が同じように，送られてきた演奏情報すべてに反応したのでは，オーケストラのような演奏にはなりませんね。各パートごとの別々の演奏情報として1本のケーブル上に送ってやる必要があるわけです。

そこで，同一ケーブル上に接続された楽器を独立コントロールする方法としてMIDIチャンネルが用意されたのです。MIDIチャンネルは，ちょうどケーブルテレビのようなもので，送信側のチャンネルと受信側のチャンネルが一致したもののみ受信されるという仕組みです。そして，MIDIで用意されているチャンネル数は16チャンネルと決まっています。

もちろん各楽器にもMIDI受信チャンネルの設定があります。そこで，いくつかの楽器を接続しておき，演奏させたいパートと，楽器のMIDIチャンネルを一致させておけば各楽器ごとに別パートの演奏ができるというわけです。

MT-32のようなマルチ音源は，複数のMIDI受信チャンネルを持っているのです。MT-32の場合，8パート＋リズムパートで計9つのMIDIチャンネルを使用でき，リセット状態ではMIDIチャンネル2ch-9chが8つの各パートに自動的に割り当てられます。そして，リズムパートは11chです。多くのマルチ音源は，もっと自由なMIDIチャンネル設定などができますが，操作はその分，複雑です。

図3　マルチ音源の例

●C-TRACE 68

より使いやすく充実して新登場

Tan Akihiko
丹　明彦

半年前に一度ご紹介した「C-TRACE 68」が，バージョンアップされて新登場。スピードアップ，操作性のアップなど，改善された部分に加えて，今回はサンプルプログラムを交えながらご紹介していくことにしましょう。

X68000用　キャスト　5"2HD版　68,000円　☎03(797)5128

お馴染みのレイトレーシングツールC-TRACEがチューンナップされて再び僕らの前に姿を現した。パッケージにまぶしく光るステッカーには，「NEW VERSION!」の文字と例のステゴちゃんがあしらってある。演算スピードも，操作性もアップ。ステゴちゃんをはじめとする豊富なサンプルデータまでついてくるときた。紹介はすでにやっているので，前と比べて変わったところを中心に駆け足でいってみよう。

速さが命！

とにかく，レイトレの宿敵は「時間」である。計算時間の壁が克服されてレイトレが見直されてきたのはつい最近の話で，それまでのレイトレは個人(もしくはヒマ人)の趣味程度にしか考えられていなかった。リアルタイムの表示(！)ができるときまで，当分レイトレソフト開発者と時間との戦いは続くだろう。

C-TRACEは今回のバージョンアップでスピードアップを果たしたが，特に数値演算プロセッサとコンビを組んだときに計算が速くなる。ベンチマークよろしく適当なサンプルを持ち出してきて，各バージョン・数値演算プロセッサありとなしの場合でそれぞれ計算させて，実際どのくらい速くなったかを比較するのもいいが，そんなことはあまり意味がない。この手のソフトは，常に最新バージョンを使うべきだし，また使っていればいいのだと思う。

数値演算プロセッサ使用時で2～3倍の高速化というが，使ってみると，確かに速くなっている。それだけで十分だ。数値演算プロセッサがない場合でも，多少は高速化を期待できるし，今月掲載されているFLOAT2＋.Xを使えばさらに40％の高速化が期待できる。以前掲載されたFLOAT3＋.Xも健在で，同程度の効果をあげている。レイトレーシングのような処理には必携のドライバといえるだろう。

こまごまとしたチューンアップ

本体のプログラムRAY.Xは，計算が速くなったということを除いては，基本的に変更はない（速くなったというのが，それ自体大きいが)。当然ながら，旧バージョンで作ったレイトレース用のテキストファイルやイメージファイルなどのフォーマットもそのままで，安心して使える。

しかし，周辺のツールやアプリケーションにはかなりの機能的な改良のあとが見られる。FUT.Xなどの画像ファイルを扱うツールでは，法線マッピングなどに不可欠な白黒ファイルがちゃんとサポートされている。これまでは自作のツールで間に合わせていたのだが，これでZ'sSTAFFでマップを描いてC-TRACEで利用することが，当たり前のようにできるようになった。

ほかにも，モデリング用のツールSPED.Xでは，セーブしておいたテキストデータを再び読み込むこともできるようになった。つまり，新規作成だけでなく，継続して編集することもできるようになっている。もともとモデリングツールとしてのSPED.Xは非常に優秀だっただけに，これは非常に大きなメリットといえるだろう。プリミティブの追加も自由自在だ。が，残念ながらSPED.Xだけでは，あいかわらず一度定義したプリミティブの削除・変更などはできないし，論理演算も扱えない。最終的にはテキストで記述するというのがC-TRACEの基本精神のように思われる。

ただし，SPED.Xで出力したファイルは，オブジェクトデータファイル（拡張子はRDT)の中に，file文で取り込める。C言語風にいえば，インクルードできる。SPED.Xで使うオブジェクトデータファイルのフォーマットはかなりかっちりと決められているので，レンダリングに必要な情報をそのままテキスト内に記述してしまうとSPED.Xに戻って再編集できない。そこで，編集するためにSPED.Xで作ったファイルはそのまま手をつけずに残しておき，描画のときだけデータとして別のテキスト中に取り込むという方式をとれば，結構対話的な環境が手に入るだろう。

マニュアルも旧バージョンに比べて親切になっている。SPED.Xの説明も丁寧にされているし，インストールの方法や，簡単な実習例にもきちんとページを割いている。なお，インストールの際には，数値演算プロセッサの有無や，拡張メモリの有無などを聞いてくる。システム構成によって処理の仕方を変え，高速化を図っているようだ。

近頃おいしいアンチエリアシング

本番の描画のときは，詳細な画像を出したくなるものだが，こういうときにはアンチエリアシングがおいしい。前に紹介したときにはアンチエリアシングは使わなかったが，あとからこれがおいしい機能だとわかってきた。

アンチエリアシングは，ピクセルの色がわりと極端に変化している個所などをさらに深く調べ，いちばん自然に見える色にする機能だ。同じ分解能でアンチエリアシングをかけるとノーマルのときより時間はか

グレードアップしたグラス

かるが，詳細な画像を出したいときは，ただ分解能を上げるならむしろ，分解能はそのままでアンチエリアシングをかけたほうが短い時間で済むし，綺麗な画像が得られる。輪郭は滑らかになるし，分解能以上の表現が可能になる。アンチエリアシングなしでやると，分解能を上げても輪郭線がガタガタとしていて，ちょっとみっともない。アンチエリアシングは，美しさの割には計算時間が節約できる，非常におトクな機能だ。

表現力はやはり凄い

さて，サンプルとしてテーブル上の3つのグラスを作ってみた。前回のグラスはあまりにデキが悪かったので，少し改良している。テーブルの石の模様（MABLE.IMG）は，Z'sSTAFFで描いた。前にX68000を描いたときのように，石の写真を持ってきてイメージスキャナで取り込むと速いし便利だが，なくてもどうにかなるものだ。ほかの構図などは，次のとおり。ここに書いた以外のことはシステムが与える初期値でいい。

view：100,50,150
target：0,30,0
zoom：60
ratio：1.5
mode：1,2,8,2

屈折を8回としたのは，ガラスを1枚透過するごとに2回屈折があるからだ。画面の一部に，2つのグラス，つまり合計4枚のガラスを透過するところがあり，屈折回数が足りないと，そこが真っ黒になってしまう。アンチエリアシングのレベルは2。なお，全部のグラスを構図に入れようと思ったので，zoom（画角）を60°としたが，そのせいで画像にひずみができてしまったようだ。

気になる計算時間のことだが，256×256ピクセルで約10時間といったところだ。屈折8回，マッピングとアンチエリアシングつきなら速いといえるだろう（数値演算プロセッサボード使用時）。

まとめ

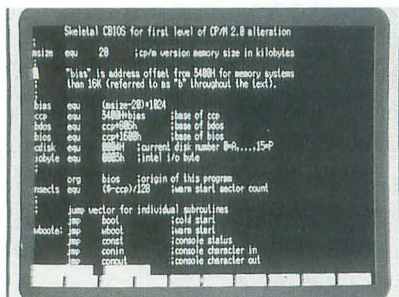
新バージョンの印象をまとめてみよう。取り立てて目新しいことはしていないが，使いやすいように整備されているし，なによりも速くなったというのが嬉しい。

ただ，Human68k（MS-DOS）の操作法やコマンドは知っておく必要があるし，オブジェクトの完全な記述はSPED.Xだけでは無理で，最終的にはテキストエディタで指定しなくてはならないので，この点では操作性をウリにした彩CRONEには一歩譲る。

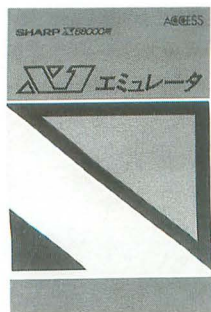
また，一度彩CRONEの強力なマクロ機能に触るとプリミティブをまとめて動かせないのも少々うざったい。テキストデータにフィルタをかけてやればいいのかもしれないが，これがSPED.X上でできれば素晴しいに違いない。さっき作ったサンプルでも，ひとつのグラスを動かすのに，グラスを構成するプリミティブをいちいち動かしている。

リアルな表現をするためには細やかな指定が，自由なデザインのためには使いやすいツールが必要だ。C-TRACEか彩CRONEかの選択のカギはこのへんにあるような気がする。デザイナーの思考を妨げない操作性と速さを持ち，かつ誰もが感心するようなリアルな表現もできるツールの完成はまだまだ先の話になるが，これらのツールをひとりでも多くの人が使い，開発者にフィードバックすれば，必ず達成されるだろう。そう期待したい。

リスト　glasses.rdt

```
  1: /*** テーブル上の3つのグラス ***/
  2:
  3: ratt
  4:         glass_r
  5:                 0       0       0
  6:                 0.1     0.1     0.1
  7:                 10
  8:                 100
  9:         table_r
 10:                 0.0     0.0     0.0
 11:                 0.4     0.4     0.4
 12:                 0
 13:                 0
 14:                 :cmap
 15:                 mable
 16:                 1.0
 17:                 pl
 18:                 -100    -100    100     100
 19:                 -100    -100    100     100
 20:         none_r
 21:                 0     0     0
 22:                 0     0     0
 23:                 0
 24:                 0
 25: end
 26:
 27: tatt
 28:         glass_t
 29:                 0.99    0.99    0.99
 30:                 1.52
 31:         none_t
 32:                 0.0     0.0     0.0
 33:                 1.0
 34: end
 35:
 36: shp
 37:     /* グラス1 */
 38:         ex1
 39:                 uc      5       -5      5
 40:                 :my     30
 41:         in1
 42:                 el      -5      5       -5
 43:                 :my     30
 44:         top1
 45:                 pl      0       1       0       60
 46:         leg1
 47:                 cy      2       -16     2
 48:                 :my     17
 49:                 *r glass_r  *t glass_t
 50:         std1
 51:                 el      30      10      30
 52:                 :my     -8
 53:         btm1
 54:                 pl      0       -1      0       0
 55:     /* グラス2 */
 56:         ex2
 57:                 el      20      30      20
 58:                 :my     60
 59:                 :mxz    20      60
 60:         in2
 61:                 el  19.5    29.5  19.5
 62:                 :my     60
 63:                 :mxz    20      60
 64:         top2
 65:                 pl      0       1       0       60
 66:                 :mxz    20      60
 67:         leg2
 68:                 cy      2       -15     2
 69:                 :my 15.5
 70:                 :mxz    20      60
 71:                 *r glass_r  *t glass_t
 72:         std2
 73:                 el      30      10      30
 74:                 :my     -8
 75:                 :mxz    20      60
 76:         btm2
 77:                 pl      0       -1      0       0
 78:                 :mxz    20      60
 79:     /* グラス3 */
 80:         ex3
 81:                 el      5       -20     5
 82:                 :mxz    50      10
 83:         in3
 84:                 el      -5      20      -5
 85:                 :mxz    50      10
 86:         top3
 87:                 pl      0       1       0       80
 88:                 :mxz    50      10
 89:         std3
 90:                 el      30      10      30
 91:                 :my     -8
 92:                 :mxz    50      10
 93:         btm3
 94:                 pl      0       -1      0       0
 95:                 :mxz    50      10
 96:     /* テーブル */
 97:         table
 98:                 bx      100     2.5     100
 99:                 :my -2.6
100:                 *r table_r  *t none_t
101: end
102:
103: log
104:         g1_1
105:                 :* in1 ex1 top1
106:                 *r glass_r  *t glass_t
107:         g1_2
108:                 :* std1 btm1
109:                 *r glass_r  *t glass_t
110:
111:         g2_1
112:                 :* -in2 ex2 top2
113:                 *r glass_r  *t glass_t
114:         g2_2
115:                 :* std2 btm2
116:                 *r glass_r  *t glass_t
117:
118:         g3_1
119:                 :* in3 ex3 top3
120:                 *r glass_r  *t glass_t
121:         g3_2
122:                 :* std3 btm3
123:                 *r glass_r  *t glass_t
124: end
125:
126: lit
127:         parallel
128:                 pa
129:                         1     -1     -1
130:                   10000
131:                         1      1      1
132:         ambient
133:                 am
134:                     0.4    0.4    0.4
135: end
136:
137: back
138:         0.1       0.1       0.2
139:
140: allend
```

世界が広がるパソコン二人三脚

Nakamori Akira
中森　章

X1とX68000の2台を使い分けることを強いられていたユーザーにとって，ファイルが共有できるエミュレータの登場はたいへん興味のあるところ。それではようやく登場の「X1エミュレータ」の持つ魅力を探ってみることにしましょう。

X68000用　5″2D/2HD版　2枚組
(専用ケーブル付き) 9,800円
アクセス　☎03(233)0200

夢のエミュレータ登場

X68000が初めて発表されたとき，CPUにMC68000を採用した本格的なパソコンが登場したということでかなり注目を集めました。当時X1turboのユーザーであった私も，X68000が発売されたらすぐに買おうと心に決めたものです。そこにはX1，X1turboと着実に実績を積み重ねてきたシャープに対する大きな期待があったからです。

最近では，初めて購入したパソコンがX68000だという恵まれた環境のユーザーも多いようですが，X68000の発売当初は，私と同様，X1あるいはX1turboからX68000に乗り換えた，というより，あえてもう1台購入するといったタイプのユーザーが多かったのではないでしょうか。このようなユーザーにとっては，X1は心の故郷とでもいうべき存在で，X68000を手にしたとき，多くのユーザーはX68000上でX1シリーズのソフトが動いたらいいなと思ったことでしょう。

しかし，X1シリーズのCPUはZ80，X68000のCPUはMC68000で，CPUには互換性はまったくありません。また，X68000のディスクドライブはX1シリーズ用の2Dのディスクを読み書きすることはできません。このような事情からX68000の上でX1のプログラムを実行させることは夢の域を出ることができませんでした。しかし，X1 LOGOやASK68Kなどの開発で有名な「アクセス」は，その夢の実現に挑戦してくれたのです。今回紹介する「X1エミュレータ」がそれで，これはX1(turbo専用のものは不可)のプログラムを，X68000の上で実行するソフトウェアなのです。

エミュレータとは

エミュレータとは，既存のものに対して，エミュレート（負けないように努力する，真似る）を行う機構のことです。またエミュレートを行うことをエミュレーションといいます。特にコンピュータの分野では，異なるCPUのマシン語の動作を真似る機構をエミュレータと呼びます。

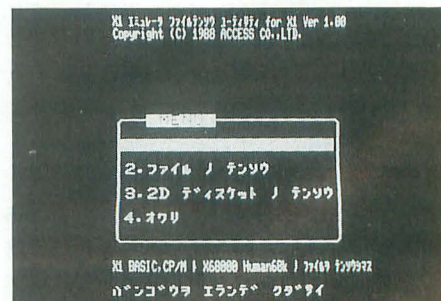
通常はハードウェアを用いてCPUの動作を真似る装置のことをエミュレータと呼びますが，ときにはプログラムによってCPUの動作を真似る場合もエミュレータ（ソフトウェアエミュレータ）と呼ぶことがあるようです。

しかし，プログラムによる場合はシミュレータというほうが一般的のような気がします。そういう点から見れば今回のX1エミュレータは，すべてソフトウェアに依存してZ80の真似をするプログラムですから，「X1シミュレータ」と呼んだほうが正確なのですが，言葉の持つイメージ的なものから「エミュレータ」としてあるのでしょう。ここの紹介でも，エミュレータという言葉をそのまま使っていくことにします。

ソフトウェアエミュレータでは，エミュレーションの対象となるCPUの持つレジスタなどのハードウェア資源を変数としてメモリ上に確保し，プログラムが読み込み解釈する命令の動作内容に従って，メモリ上の変数間でデータの移動を行うことでCPUの動作を真似るようになっています。たとえば，Z80のエミュレータのプログラムが，

21H　32H　12H (LD HL，1234H)

という命令コードを読み込んだら，それがHLという内部レジスタに定数をロードする命令であることを認識し，その結果，HLという変数に1234_Hという値を書き込みます（これはあくまでも原理であって，X1エミュレータがこうなっているかは不明）。

実際のZ80ではこれら一連の動作は，ハードウェアによって一瞬のうちに処理されてしまいますが，プログラムでエミュレートする場合はいくつものステップが必要になります。このためエミュレータでの動作は実際のCPUよりも実行速度が遅くなってしまいます。本当に「負けないように努力する」という意味がぴったりですね。なお，X1エミュレータでは実際のX1での実行に比べて，3～5倍程度の時間でプログラムの実行ができるようです。

X1からファイル転送

X1エミュレータの概要

それではX1エミュレータの概要について説明しましょう。X1エミュレータは，Human68kのディスク上に作られたX1用5インチ2Dディスクと同じイメージを持つ，320Kバイトのファイルを X1の仮想的なディスクドライブと見なしてエミュレーションを行います。そしてX1エミュレータで仮想的に実現したX1はこの仮想ディスクドライブ用ファイルから起動するようになっています。このためBASICやCP/Mなど，X1を起動するためのシステムディスクをあらかじめ仮想ディスクドライブ用ファイルに変換しておかなければなりません。

仮想ディスクドライブ用ファイルは，X1エミュレータに付属しているファイル転送ユーティリティを使って作成することができます。Human68kのディスク上のどの仮想ディスクドライブ用ファイルがX1のどのディスクドライブに対応するかは，X1エミュレータ起動時に指定するようになっています。このときドライブの指定は，FD0からFD3までの4つを仮想ディスクドライブ用ファイルに割り当てることができます。

X1のプログラムは通常ドライブ0から起動しますから，BASICやCP/Mのシステムが入った仮想ディスクドライブ用ファイルをFD0に指定してX1エミュレータを起動すれば，X68000上でX1のプログラムを実行することができるようになるのです。

さて，X1のプログラムの実行が開始されるといきなりX68000の画面上になにか枠が描かれます。最初はなにかわかりませんでしたが，どうやらこれはテレビの画面の絵のようです。つまり，X1のプログラムはX68000の画面に描かれたテレビの画面をX1の画面として動作するのです。X68000とX1では画面のドット数が異なるため（当然X68000のほうが多い），X68000の画面でX1のプログラムをそのまま実行したのでは，画面に余白ができて周りが寂しくなってしまいます。X1エミュレータではその画面の余白にテレビの絵を描いて，にぎやかさを出そうとしているわけです。この画面に描かれるテレビの色は赤ですが，ついでに銀色とか白色（！）などにも変更できるようになっていたらもっとウケたのではないでしょうか(だって，私の最初のX1は銀

懐しの(？)New BASIC

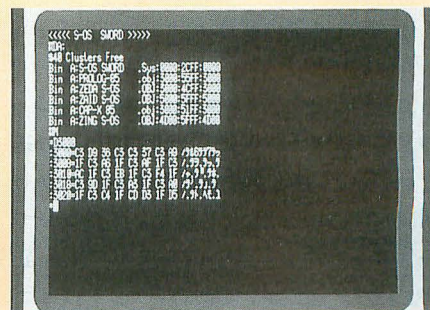
“SWORD” も走る

色だった)。

それはともかく，このX1エミュレータで私の持っているX1のアプリケーションプログラムを実際に動かしてみたところ，Hu BASIC，CP/M，ランゲージシリーズ，X1 LOGO，S-OS “SWORD” は動くようでした。しかし当然ながら，市販のゲームなどのプロテクトがかけられているものについては，仮想ディスクドライブ用のファイルが作成できないため，X1エミュレータで実行することはできませんでした。

表1に実行可能なアプリケーションソフトの一例を紹介しておきます。ただし，これらのアプリケーションソフトのなかでも，タイミングなど一部ハードに依存しているような部分があるものに関しては，正常に動作しないようです。

ところで，X1エミュレータのエミュレーション動作はZ80の命令をMC68000の命令により1ステップずつソフト的に実行することで実現されます。しかしそれだけではZ80をエミュレートしただけで，X1というパソコンをエミュレートしたことにはなりません。そこでX1をエミュレートするためには，X1のI/O機能（VRAMのアクセス，ディスクのアクセス，メモリバンクの切り換えなど）をエミュレートする必要があります。

当然，X1エミュレータはX1のI/O機能をエミュレートしていますが，残念なことにI/O機能のすべてをサポートしているわけではありません。このため，X1エミュレータでサポートしていないI/O機能を使用するプログラムは動作しませんから注意が必要です。参考までにI/O機能のサポート状況を表2に示しておきましょう（マニュアルより転載）。この表を見ながら，自分の持っているX1のアプリケーションプログラムがうまく動作するかどうかを判断してみてください。

ファイル転送ユーティリティ

おそらく，X68000でソフトウェアによるZ80のエミュレータを作るという考えは，多くの人が持っていたことでしょう。しかし，エミュレータを作るうえで最大の問題点となるのは，X1とX68000のディスクドライブの間に互換性がないことでした。このため，エミュレータを作ろうと思った人のほとんどは挫折していったのではないでしょうか。

ところが，X1エミュレータでは，X68000のディスクドライブでX1のディスクを読むという考えをあっさりと捨てて，X1のディスクのイメージをそのままHuman68kのファイルとして扱うという手法を採用しました。そしてそのために専用のファイル転送ユーティリティを用意してくれました。ここではこのファイル転送ユーティリティについて説明しましょう。

X1エミュレータに付属しているファイル転送ユーティリティは，X1のプリンタ用ソケットと，X68000のジョイスティック用ソケットの間を付属の専用転送ケーブルで接続してファイルの転送を行います。転送の種類としては，

表1　実行可能なアプリケーションソフト

HuBASIC
X1 CP/M
X1 LOGO

X1 CP/M用ランゲージシリーズ
- APL
- LISP
- COBOL
- C
- FORTH
- FORTRAN
- PASCAL

注）一部制限される機能があります

1) ディスクの転送

2) BASICファイルの転送

3) CP/Mファイルの転送

の3種類があり，X1からX68000への転送だけでなく，2)と3)についてはX68000からX1という逆の転送もできるようになっています。1)のディスクの転送が，Human68k上のディスクに仮想ディスクドライブ用ファイルを作るための転送で，このソフトの核となっているところです。プロテクトのかかっていないX1用のディスク(5インチ2D)ならば，基本的には支障なく転送することができます。

2)と3)の転送は付属の機能といったところで，BASICやCP/MのディスクとHuman68kのディスクとの間でファイル単位の転送をすることができます。CP/MにはSmall-CやXLispなど，ソースプログラムが公開されているパブリックドメインのソフトウェアがたくさんありますが，それらを簡単にHuman68kに持ってくることができるようになります。

マニュアルによると，ディスクおよびファイルの転送ではデータを圧縮して転送しているため，ディスクに記録されている容量が少ないほど高速な転送ができそうです。実際ファイルの転送時間には転送するディスクの種類によってかなりばらつきがあるようで，私の経験でいくと1枚のディスクの転送にかかる時間は，20〜30分と考えておけばよいでしょう。

X1エミュレータの限界と感想

今回のエミュレータを評価するにあたって，昔のX1のアプリケーションプログラムを探してみたのですが，ほとんどがテープ版でディスク版のものは少数しかありませんでした。それもそのはず，私の初代X1にはディスクドライブが付いてなかったのです。ディスク版のアプリケーションソフトがポピュラーになるころにはX1turboを購入していましたから，X1専用のディスク版のプログラムはほとんど持っていなかったのです。

このような状況は，私だけでなく多くのユーザーの方も同じようなものだと思います。つまり，X1のアプリケーションソフトのほとんどはテープに入っているのです。しかし，X1エミュレータはカセットテープ関係の機能をサポートしていませんから，結局私の場合，数的には動くソフトウェアよりも動かないソフトウェアの数のほうが多いということになってしまいます。このところがX1エミュレータの限界といえるでしょう。

しかし，これについては，X1エミュレータが手を抜いたというよりも，X68000にカセットインタフェイスが付いてないのでしかたがないかもしれません。まあ，BASIC(New BASICでは漢字表示ができなかったが)とCP/Mと“SWORD”が動くだけでもよしとしましょう。

特に，付属のファイル転送ユーティリティはなかなか便利です。従来でも，CP/MのファイルならばRS-232Cを介してなんとかX1からHuman68kに転送することができましたが，BASICのファイルをHuman68kに転送するのはたいへんでした。それが簡単にこのソフトを使えばできるのです。テープ版のソフトが動かないという不満が残るとしても，誰でも簡単に扱える操作性を持ち，9,800円という手頃な価格とともに，ファイル転送ユーティリティを手に入れることだけを考えてみても，買う価値はあるのではないでしょうか。

もちろんCP/Mも大丈夫

表2　X1エミュレータのサポート状況

機　能	サポート	備　考
BASIC ROM	×	
IPL ROM	○	外部コール可能なルーチン＝注1参照
メモリバンク	○	
ユーザーI/O	×	
外部メモリ	×	
フロッピーディスク	○	2Dのみ
テキスト画面	○	横80/40　縦25/20/12/10＝注2参照
テキストVRAM	○	直接R/W可能
アトリビュートVRAM	○	直接R/W可能
テキストプライオリティ	×	
ROM CG	○	
PCG	○	X1標準の方法でのアクセスのみ
漢字ROM	△	テキスト表示のみ可能（直接アクセス不可）
G-RAM	○	直接R/W可能
G-RAMバンク	○	表示用/アクセス用　200/400lines
カラーパレット	○	セットのみ（読み出し不可）
タイマ	×	セットは可能（タイマ動作はしない）
日付/時間	×	セットは可能（刻時動作はしない）
プリンタ	×	
カセットI/O	×	
PSG	×	
ジョイスティック	×	
TVコントロール	×	

表のサポート欄：○＝サポートあり　×：サポートなし　△＝制限つきサポート

注1)　IPL ROMのうち，プログラムからのコールをサポートするのは
1)　IPL起動
2)　DISKロード
3)　1文字出力(コントロールコードはCR，TAB，^Y，^Lのみ対応)
4)　スクリーンクリア
5)　サブCPUチェック
の各ルーチン。

注2)　画面モードを切り換えると，エミュレータの外枠（テレビの絵）を描き直す。

注3)　表中，「直接R/W可能」というのは，たとえばVRAMであれば，「X1のVRAMに直接書き込むプログラムでも，エミュレートして画面表示が可能」という意味であり，X1のプログラムが直接にX68000のVRAMをアクセスする，ということではない。

未知の領域に挑む職人芸の世界(中編)

Kuramochi Ryoichi
倉持 亮一

今月はこれまでの職人芸の実績を振り返りつつ，果たしてこのままでいいのかと，数多くの疑問を提示しながら話は進みます。さて，次世代のゲーム界に新風を巻き起こすのはいったい誰か。いよいよ核心に向けて迫っていきます。

先月号に引き続きアーティストとアーティザンの話である。とにかく，私はやたらめったらと大風呂敷を広げてしまったので，かなり話の展開が複雑怪奇になってしまった。できれば，もう一度先月号を復習してから読んでいただければありがたい，と，わがままをいいつつ，いきなり話は本題へと進む。

とにかくゲームには，アーティザン（職人）が作ったアーティザナル（職人的）なゲームと，アーティストが作ったアーティスティック（芸術的）なゲームの２つのタイプがこの世に存在する，というのが前回のメインテーマだった。そこで今回は，いまアーティザナルなゲームがもてはやされている原因を，大胆にもゲーム以外の娯楽のジャンルまで踏み込んで解明するところから始めようと思う（ああ，とどまることを知らない私の大風呂敷）。

楽しくなければゲームじゃない

広辞苑で「娯楽」という言葉を調べてみたら，「人の心を楽しませ，慰めるもの」とだけ書いてあった。あれまあ，ずいぶんと簡単に定義してくれちゃったこと。私は根っからの文系人間で，しかも学校のゼミでは「比較言語学」なるものを専攻したせいか，話のネタを考えるときには，ふだん自分たちが何気なく使っている言葉の意味を調べ，自分が思っていた意味とのギャップを見つけては，そこからどんどんと考えを膨らませていく手法を取ったりしている。余談になるが，実は今回のテーマも，テキスト中にあった「artisan」という言葉を偶然見つけたことから始まっている。

なんにしても，こんなにあっさりと定義されてしまっては，話の進めようがないじゃないの。でも，広辞苑に書かれている「楽しい」とは，いったいどういう状態を指すのだろうか。

とにかく人がワクワク，ドキドキ，ハラハラ，ガハハハ（これ笑い声ね）という状態にあれば，「楽しい」ということになるのだろうか？

日本に昔からある娯楽のひとつである将棋の場合を考えてみると，そばで見ている限り少なくともハラハラ，ドキドキという感じではない。どちらかといえば，「イライラ」というイメージが強い。プロの棋士ともなると，対局中は制限時間に追いまくられ，胃に穴が開くほどイライラしているようにも見える。

しかし，どんなに対局中にイライラしようとも，勝てばとても嬉しいし，また楽しいはずである。娯楽に勝負がからんでくれば，途中どんな状況にあったとしても，最終的に楽しいハイな気分になればすべて丸く収まるものなのである。しかし，途中でイライラさせられるわ，勝負には負けるわ，という悲劇を味わったとしても，「もー，イヤだっ！　二度とやるもんか」とかなんとかいっちゃって，机を引っ繰り返して星一徹のように暴れ回るかというと，そんなことはない。

将棋に限らず，麻雀，トランプ，ゴルフにパチンコと，一度負けたからといって頭を丸めて身を引くような殊勝なヤツはどこにもいない。あえて何度も挑戦することになるのだ。悲しい人間の性である。

つまり娯楽というものは，常に「楽しい」状態がなくても成立するものだ。言い換えれば，娯楽においてはイライラも，難しくて頭を悩ませることも，負けることもすべてが楽しさにつながっているのである。逆をいえば，そういったことのすべてが，楽しさに変わらなければ，娯楽とは呼べないのだ。

たとえば，シミュレーションゲームなどをプレイしているときは，思いっきり頭を悩ませるが，プレイヤーはその過程を楽しんでいる。これは将棋のイライラに近い感覚だと思う。じゃあ，受験勉強も同じじゃないかって？　人の話を真面目に聞く気のない子は，偏差値分析プログラムでも作ってひとりで遊んでなさい。

娯楽とマクドナルド

よく考えてみると，どうやら娯楽には２つのタイプがあるように思う。ひとつは，直接なにかを見たり，その場を共有することによってダイレクトに楽しさを感じるもの。もうひとつはムズイ，キツイ，頭がウニになるといったような体験をしていても，心のなかではジワジワと楽しさがアメーバのように広がっているタイプである。

前にも述べたように，アーティザンは，プレイヤーを楽しませることを念頭に置いているから，当然，アーティザナルなゲームというのは，前者のタイプだ。イースやドラクエシリーズなどはその典型である。きちんと１本のストーリーが設定されているRPGの常として，途中のストーリーがダレないようにいくつかの難関（小ボスとか，奥深いダンジョンに隠されたアイテムなど）が用意されてはいるが，その難関へプレイヤーが到達するころにはチョット苦労するだけで突破できるようにプレイヤー側のレベルが上がっているように，うまくゲ

ームバランスがとられているのだ。

このことは，ゲーム中に8つの徳のレベルを下げないように細心の注意を払わなければウルティマⅣなどと比べたら，なんと楽チンで楽しいことか。もちろん，楽しさをダイレクトで提供してくれるゲームは，RPGに限らない。昨年，X68000に発売された「熱血高校ドッジボール部」などはその代表的な例である。とにかくプレイヤーはなんにも考えずに，ボコボコとボールを相手にぶつけることだけしてればいい。それでもプレイ中の血圧上昇率は，野球ゲームの比ではない。それほどダイレクトな楽しみを与えてくれた。

つまり，楽しさをダイレクトに伝えてくれる娯楽というのは，マクドナルドみたいなものだ(さあ，いつもの強引な展開が始まったので，置いてかれないように付いてきてね)。受け手側がイヤな気分にならないようにサービスを尽くし，また飽きられそうになったら，サンキューセットやいろいろなオマケを付けた嬉しい企画で，受け手側に刺激を与えつつ引き付けておくのである。

また，私たち受け手側もそのサービス精神に心よく反応してしまい，店に足を運んでは，「いらっしゃいませ」のカワイイ笑顔に釣られて，胸やけ覚悟でポテトのＬサイズをひとつ余分に頼んだりするのであった。ダイレクトな楽しみを得るためには，多少の出費は覚悟しなければならない。

気楽にトレンディ

マンガの世界を覗いてみると，週刊『少年ジャンプ』の発行部数が500万部を突破したそうだが，若い社会人がマンガ雑誌を読むというのは日本特有の現象らしい。欧米では，マンガは子供の読み物だと相場が決まっているらしく，あちらのマスコミなどは，日本のサラリーマンが通勤電車のなかでマンガを読んでいる写真を載せて，奇妙な現象として取り上げたりしている。

まあ，欧米では大人が読めるようなマンガがあまり存在していないこともあるのだろうが，日本のこの現象は，マンガのようなより気楽に楽しめるものを求めている若い世代の，ひとつの姿であると私は思っている。

そして，文庫本よりマンガのほうに流れている現状というのは，これは日本人が「精神的な気楽さ」を求めるようになってきたことの現れだと思う。昨今の健康ブームに乗って，テニスクラブやアスレチックジムの大繁盛を見てもわかるように，結構，積極的にあちこちで励んでいて，軽く汗を流す程度のスポーツは抵抗なく受け入れられている。しかし，空手とか相撲などのド根性的スポーツが流行しているわけではない。本を読むことにしろ，スポーツにしろ，努力と根性を必要とするようなことは敬遠され，みんな精神的苦労を避けて通るのが，いまはトレンディ（流行，傾向）なのだ。

ものぐさなあなたに

ご存じだとは思うが，RPGというものは，もともと複数の人間がテーブルを囲み，会話形式で進行させるものである。テーブルトーク型RPGで有名なものに，D&Dがあるが，こういったものにはモンスターの種類とデータとか，魔法の種類とその効力，戦闘方法など，とにかくゲームシステムのごく基本的な部分だけが作られていて，実際にプレイするうえでのシナリオとか，活動する世界とかはゲームマスターと呼ばれる役割の人間が設定する。

そうして，ゲームはプレイヤーが起こしたいと思う行動をゲームマスターに伝え，無事に前に進めたとか，モンスターに出会ったなどの結果をゲームマスターがプレイヤーに伝える。これは一種の高等なゴッコ遊びのようなものだが，アメリカではかなりポピュラーな遊びとなっている。確か，映画「E.T.」の冒頭にも子供たちがこのゲームに興じているシーンがあった。

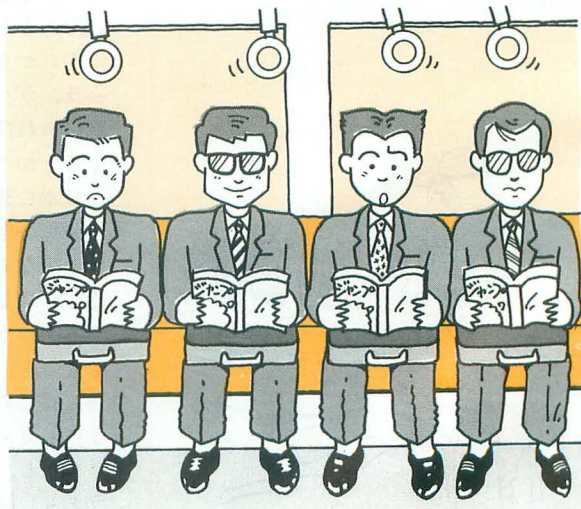
とにかくこれは，人間同士の遊びだからプレイヤーはどんな行動もとれるし，ゲームマスターの裁量によっては想像もつかない展開になったりするところが楽しい。

しかし，プレイする側にもかなりの努力を要求されるのがこのゲームの特長で，特にゲームマスターはとんでもない労力が必要となる。ゲームマスターは，プレイ中のゲームにおける神様のようなものだから，ある程度つじつまの合った世界を創っておかなければいけないし，プレイヤーたちのさまざまな行動にもきちんと対処しなければならない。ここで，プレイヤーの突然の行動に対して，「そんなこと，できるわきゃないよ」なんて平気でいっているようなら神様失格，即ゲームマスターも失格である。

プレイヤーのほうも，ゲームの状況を自分なりに想像して広げていかないと，すんなりとはゲームにのめり込むことができなくなる。日本にもテーブルトーク型RPGファンは少なからず存在するようだが，広く普及しているといえる状況ではない。テーブルトーク型を本当に楽しむためには，それなりの気概と柔軟な頭，思い込み，そして十分な時間が必要となるからだ。

これに比べれば，コンピュータのRPGなんかは気楽なものである。必要な世界やシステムはあらかじめ設定されているし，ゲームの進行上プレイヤーが行わなければならない処理はみんなコンピュータがやってくれる。しかも，グラフィックを使ってその世界を視覚的にも見せてくれるから，プレイヤーは余計なイマジネーションを働かせずにすむ。そこへもってきてアーティザナルな性質をはっきりと持ったRPGの登場である。このタイプのRPGは至れり尽くせりだから，プレイヤーはただプログラムとして用意されたシナリオに乗っかっていればいい。あとはディズニーランドのジャングルクルーズのように，受け手側はただ，ワクワク，ドキドキしていればいいのである。これは，ワクワクよりもイライラが先に立ってしまうように思える最近のシミュレーションやシューティングも同じことがいえそうだ。

これでいいのか？

アメリカでは「バランス・オブ・パワー」というちょっと毛色の変わったシミュレーションがよく売れたという。このゲーム，かなり大ざっぱにいえば「世界情勢をコントロールする」のが目的で，戦争状態になったらプレイヤーの負けというもの。これまでのシミュレーションの常識を覆すタイプのゲームといえそうだが，コツを飲み込むまではゲームを先に進めることすら難しく，すぐに核戦争が始まってしまって「ゲ

ームオーバー」という，かなりアーティスティックなゲームらしい。最近，PC-98にも移植されたが，やはりヒットチャートの上位を見てみると，もはや定番商品となってしまった「大戦略」や「信長の野望」が名を連ねている。

一方，戦国シミュレーションの大御所である光栄も，少し前に「思想統一シミュレーション」と斬新なコンセプトを持った「維新の嵐」を発売したが，売れ行きではやはり「大戦略」や「信長の野望」というスタンダード商品には勝てなかったようだ。これは，ゲームソフト買う側からすると「どうせ高いお金を出して買うなら冒険するより安全牌を」の精神がモロに出てしまった結果なのだ。

娯楽は人を楽しませることが大原則だから，お笑いタレントのさんまみたいに，その形態がダイレクトに楽しさを伝えるものでも，また横山やすしのようにアクのある強い個性が表に出ていて，そのうちに楽しさがわかってくるようなタイプでも，このどちらでも私はいいと思っている。このまま，「精神的な気楽さ」を求めるトレンド（もはやナウイなんぞは死語なのだ）に流されて，アーティザナルなゲームに走ってしまうことに，私は一抹の不安を感じてしまうのだ。

いまも変わらぬ伝統の味

くどいようだが，アーティザンは職人芸を駆使してアーティザナルなゲームを作る。その職人芸というのは，ある日突然にポコッと現れるような簡単なものではなく，昔からの技術にさらに磨きをかけ，洗練されたものを形にしようとする。さらにいうと，いま現在形成されている市場にポイントを絞ってアレンジし，よりよいものを完成させる。ここに職人芸の腕の見せどころがある。

たとえば，先月も登場させた映画の「スターウォーズ」である。ベースとなっているのは，昔からある「剣と魔法の物語」。しかし，この映画は，映画史上に残るほど数多くの観客を動員する魅力を持っていた。それは，いつの時代も子供たちが持っている好奇心を満たしてくれるには，「剣と魔法の物語」は，とても重要なファクターだからだ。

そこに，現代風にアレンジするためのスピルバーグとルーカスという2人の職人芸が加われば，これはもう，鬼に金棒，X68000にアフターバーナーの世界である。さらにそこへ，「立場の逆転」または「最後の一発大逆転」などの手法を盛り込んでストーリーが進展すれば，見ている側からすれば，たとえ最後には正義が勝つとはわかっていても，勝負の行方から目を離せなくなってしまうものなのだ。

ゲームでも，ボールを投げる・取るによって攻撃と防御が瞬時に一転するドッジボールや（熱血高校とちゃうよ，これは），相手から奪った駒を自分の持ち駒として使える将棋，そして，最後の1個が勝負の行方を左右するオセロなど，数え上げればキリがない。ビデオゲームでは，最初は逃げ回る一方だった「ヘッド・オン」が，やがて「パックマン」となり，一発逆転タイプとして人気を集めた。

娯楽における職人芸というのは，人を楽しませる手法として築き上げられてきたものだから，絶対に必要なものである。しかし，結局は昔ながらの手法をアレンジしたものともいえるわけだから，「いまも昔も変わらない」という手焼き煎餅の宣伝コピーみたいな手法でもある。

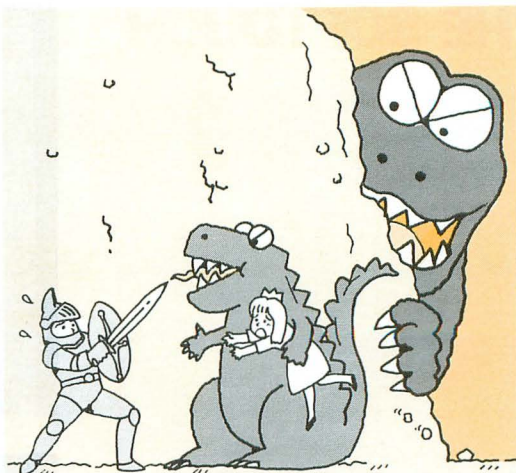
だから，コンピュータのゲームまでもが，世の中のトレンドに迎合してしまって，アーティザナルな手法ばかり駆使するようになってしまうと，これはちょっと考えものである。

一歩間違えばただの化石

その起源を古代インドの「チャトルアンガ」というゲームに求められるという，日本の将棋（今回は将棋ネタが多いから，思わず調べてしまったノダ）。西洋のチェスがそのままの形を残して定着したのに比べて，奪った駒を使えたり，相手陣地に入れば金に変わってしまうというアーティザナルな手法を取り入れて，ほとんど完成されてしまった感がある。おかげで，発展も進化もなくなってしまった。

とりあえず，挟み将棋や回り将棋，飛び将棋などのバリエーションは考え出されたものの，結局は将棋盤と駒の世界から飛び出すことはできなかった。将棋倒しや将棋崩しという掟破りのパターンもあるにはあるが，本場のドミノ倒しなどと比べると，スケールでは圧倒的に負けている。

それでも完成されたゲームとしての将棋は，これからの将棋人口という点では安泰だろうし，そうかといって，小学校の授業に組み込まれるほど劇的な飛躍を遂げるものでもない。もはや，将棋は将棋でしかない。

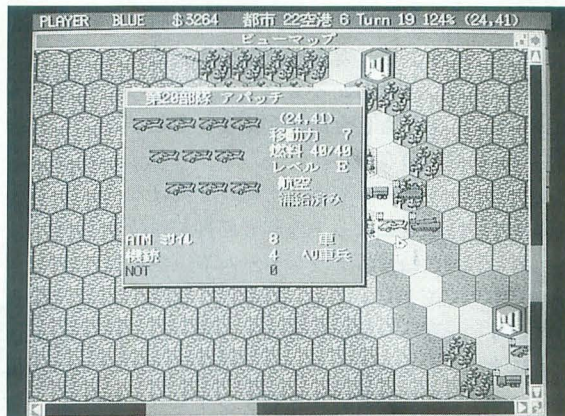
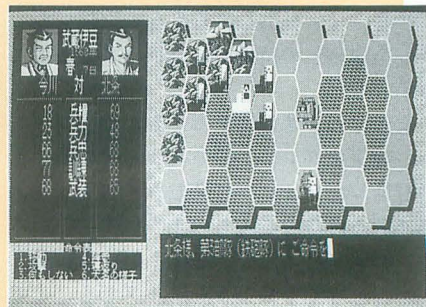
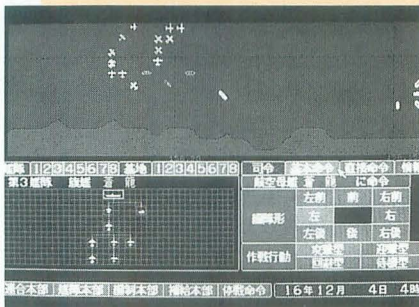
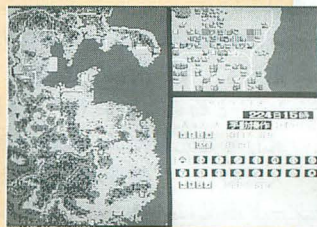
私としては，コンピュータゲームが職人芸に頼ってもらってばかりでは困ってしまう。ヒロイックファンタジー（英雄が活躍するファンタジー）で，昔からよく使われる手法に，「お姫様を捜しに敵の本拠地に殴り込んだら，すでに親玉が連れ去ったあとだった」とか，「敵のボスだと思っていたら，実は大ボスがさらに控えていた」といった，「大どんでん返し」（ちょっと待ったコールではない）がある。

これは受け手を驚かせたり，感動させたりするのには一番都合のいい手法だが，RPGやシューティングなど，あちこちに氾濫してしまい，もはや意外性の常套手段と化してしまった。このような手法を否定するつもりはまったくないが，人を楽しませるためにいつまでも絶対に必要なファクターだと思っていると，そのうち進化や発展の可能性が閉ざされてしまい，将棋のように文化の片隅でご隠居生活を送るハメになりかねない。

To be continued

さて，大きな風呂敷をそろそろたたみ始めようとする私の謙虚な姿勢が見え始めたところで，残念ながらスペースが尽きてしまった。うーん，困った。本当なら2回で終わるはずだったのが，もう1回あったりする。とにかく，来月のゲーム特集で私は必ず決着をつけるつもりだから，最後までお付き合いいただきたい。もし，このままひとりでズルズルとやっていたら，「コイツはOh！XのVか」などと陰口を叩かれかねない。さすがにこの私も，「V2」全10巻まではこの身が持たない。

来月は，アーティスティックな要素を重点的に掘り下げ，そこからゲームに対してはどのような「刺激」をもたらすのかを，じっくりと考えてみたいと思っている。ではまた来月。

SOFTOUCH PRO-68K

ライトニングバッカス
ソフトでハードな物語2
ホテルウォーズ
蒼き狼と白き牝鹿・ジンギスカン
デジタルサウンドシステムDiSS-P
C&プロフェッショナルパッケージ
プログラマーズ・ツール・キット

これがX68000用に発売されたシミュレーションゲームの数々。ただし，左下の大海令は3月発売予定なので，写真は98版のものです。あしからず

OS-9も発売されて，ますます活気づいてきたX68000のソフトウェアですが，今月はまずはシミュレーションゲームのお話。ようやく発売されたシステムソフトの**SUPER大戦略68K**を筆頭に，これから発売されるラインアップといえば，ボーステックの**ホテルウォーズ**，光栄の**蒼き狼と白き牝鹿・ジンギスカン**，アートディンクの**大海令**，そしてジー・エー・エムの**太平洋の嵐DX**，日本コンピュータシステムの**ライトニングバッカス**に**ヒストリーオブエルスリード**といった，豪華大作シミュレーションゲームが目白押しで登場してきます。

これまで，代表的シミュレーションといえば，六角ヘックス相手に，ジリジリと兵力を見比べながらの陣取り合戦といった，戦略シミュレーションのイメージのものでした。しかしX68000では，半年ばかりの間に集中していろいろと発売されたせいもあるのでしょうが，A列車やホテルウォーズ，ウォーニング，そしてコムパックから発売されたばかりのプロダクション・マネージャーといった，経営シミュレーションのジャンルが充実しているのが特徴なのです。

ガンガンバリバリ最新兵器が火器にものをいわせてガンバル戦争ものもいいけど，まずは知力を駆使して金儲け。やはり男たるものは金儲けに夢を懸けて，大陸横断鉄道の社長をやる，またはヨーロッパ中のホテルのオーナーになる，さらには自分がオーディションで見つけ出したタレントを売り出してマージンでがっぽり儲けるなど，ディスプレイのなかに美しい銭の花を咲かせることこそ，シミュレーションのもうひとつの醍醐味なのです。

特に，今月のSPECIAL REVIEWにも登場しているウォーニングなんかは，実際にプレイしてみると，宇宙空間で貿易するという独自のシステムを構築しようという努力のあとが見えてくるようで，とても好感が持てます。しかし，荻窪圭氏も記事中に書いているように，新しくて完成度の高いシミュレーションというものは，シナリオ中に関わってくるさまざまな要素とのバランスの取り方が非常に難しく，うっかりいろいろな要素を取り込んでしまうと，あちらを立てればこちらが立たずで，共倒れという結末を招きかねないのです。シミュレーションゲームの材料といえば，身の回りにいくらでもゴロゴロしている可能性が高いだけに，より身近で熱中できる材料というものを見つけ出して，楽しませてほしいものです。

さて，日常生活とシミュレーションという，なかなかに遊べる話題のあとは，OS-9を中心としたX68000用ツールの新製品情報です。

OS-9では，今月ご紹介のC&プロフェッショナルパッケージとプログラマーズ・ツール・キットのほかに，同じくマイクロウェア・ジャパンから，Cコンパイラ対応の**ソースデバッガSRCDBG**と**ネットワークシステム**，さらにはファイルアクセスの実行時間短縮を図れる**Disk Squeezer**などが続々と登場します。

また，ちょっと毛色の変わったところでは，MS-DOS上で開発したプログラムをHuman上で走らせるための，**X68000用クロス開発ツール**（20,000円）が，システムショップ・ハド

ソンより２月末に発売になります。このクロス開発ツールの詳細については，来月詳しくお届けする予定ですが，これまで，このような開発支援用ツールとはあまり縁のなかったX68000。このツールの登場によって連係プレイが可能になってくれれば，ますますX68000の開発環境は奥行きを増すばかり。さらには，ソフトハウスさんにとっても強力な味方となってくれるはずですから，X68000用のソフト開発にも威力を発揮してくれることでしょう。

そのほかには，すでに広告などでご存じのように，Z'sSTAFF のバージョンアップ版が間もなく登場します。このあとさらにいくつかバージョンアップされて話題を呼びそうな商品も控えているので楽しみにしてください。

ところで，この３月号が店頭に並んでいるころにはもうすでに他社のニューマシンの仕様が発表され，話題となっているかもしれません。それと同時にどのようなソフトウェアが登場してくるのかも大いに興味のあるところです。こちらも負けないよう最新情報をお届けしていきたいと思っていますから，皆さんも応援してくださいね。

X68000ソフト&ツールズ

☆……１月30日現在発売中　★……近日発売予定

★ライトニングバッカス

PC-88版で好評だった，ドラマチックシミュレーションゲーム「ライトニングバッカス」のX68000版が登場する。このゲームは，銀河系の彼方，太陽系マグダリア第７惑星を舞台に展開される８つの戦闘シナリオを制覇していくのだが，各シナリオは持久戦や防衛戦，奇襲戦など実にさまざまな設定のもとに展開されるため，プレイヤーはシナリオが変わるたびそのシーンに応じた戦略を駆使して戦いに臨むことになる。今回のX68000への移植にあたっては，他機種になかった縮小マップモードの追加や，ハードの機能をフルに活用したグラフィック，音楽など，かなりグレードアップされての登場だ。

X68000用　5″2HD版２枚組　8,200円
日本コンピュータシステム　☎03(486)6588

★ソフトでハードな物語２

昨年の10月号でもご紹介した，あの愉快なノリのアドベンチャー「ソフトでハードな物語」の続編が発売される。前作は，社長が倒れてしまったソフトハウスを，その息子が勇猛果敢(？)に立て直してしまうという内容だったが，今回は，日本全土を襲う連続ソフトハウス倒産事件をきっかけに，前回大活躍した大学生・西城宏志を中心にお馴染みモカシステムの面々が事件の解明に乗り出すというもの。とにかくこの２では，謎の美女の登場，囮捜査など，ちょっぴりシリアスな要素を加えて，そこにいつもの楽屋落ちネタが嵐のように飛び交うという。いずれにしても，コミカルな笑いとともに誰もが楽しめる作品となりそうだ。

ソフトでハードな物語２

ホテルウォーズ

X68000用　5″2HD版５枚組　7,800円
システムサコム　☎03(635)5145

★ホテルウォーズ

古くからの歴史を誇るヨーロッパを舞台に，ホテル経営を手がけ，定められた期限内にどれだけ数多くのホテルを建設し売り上げを伸ばしていくかが勝負のこのゲーム。建設地区の選択からスタッフの雇用，株の売買など，プレイヤーの経営センスによって状況が左右される楽しさは，もうすでにX1版などで実証ずみ。今回のX68000版では，全面的なグラフィックの描き直しや，ホテル完成時などのイベントに合わせた新しいカットが追加されるなど，さまざまな工夫も加えられている。また，このホテルウォーズは最高４人までプレイできるので，仲間を集めればボードゲームのような楽しさを味わうこともできる。

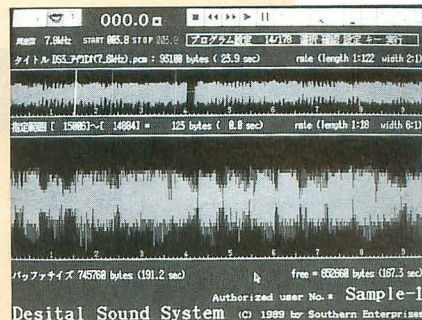
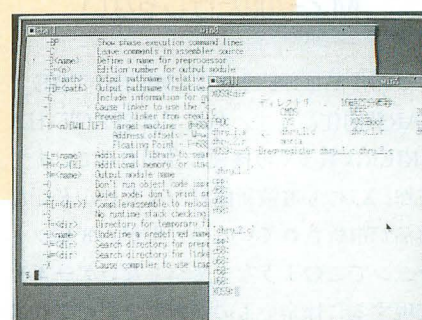
X68000用　5″2HD版２枚組　9,800円
ボーステック　☎03(708)4711

★蒼き狼と白き牝鹿・ジンギスカン

光栄の戦国シミュレーションシリーズが，舞台をユーラシア大陸に移して楽しめるのが，この「ジンギスカン」。このゲームは，昨年MZ-2500や X1に発売されて好評だったが，X68000では「信長の野望・全国版」に続く第２弾の作品となる。内容は，モンゴル帝国のジンギスカン，イングランドのリチャード１世，ビザンツ帝国のアレクシオス，日本の源頼朝の４人の英雄のなかからひとりを選択し，モンゴル平原統一かユーラシア大陸統一を目指すというもの。シナリオがスケールアップされた分，外交や同盟国の情報収集などさらに緻密な計画が必要とされるようになっている。また，光栄のオリジナル曲が収録されたミュージックテープ付きのサウンドウェアシリーズは，今後すべてCD付きに変更される予定。

X68000用　5″2HD版３枚組　9,800円
CD付きサウンドウェア　12,300円
光栄　☎044(61)6861

☆デジタルサウンドシステムDiSS-P

X68000のサンプリング機能を生かして遊べる，デジタルサウンドシステムが新しく発売された。このツールは，X68000を使ってサンプリングした音を最大90秒（1MB増設時は最大220秒）まで連続録音が可能なほか，波形グラフによる編集機能や，短音節合成によるスピーチ機能など楽しく使える豊富な機能を備え，また，できる限りマニュアルレスで使えるよう，操作方法もよりシンプルな設計となっている。ただし，このソフトの販売は通販のみとなっているので，購入希望者は下記の連絡先に，直接問い合わせてほしい。

デジタルサウンドシステムDiSS-P

C&プロフェッショナルパッケージ

X68000用　5″2HD版　7,800円
サザンエンタープライズ　☎03(787)3932

★C&プロフェッショナルパッケージ

OS-9/X68000上でのプログラム開発に必要不可欠となってくる，C言語。そのCにさらに支援ツールがパッケージされて発売される。このパッケージには，K&R及びUNIX Ver.７準拠のCコンパイラのほか，OS-9/X68000専用ライブラリ，マクロアセンブラR68，リンカL68，シンボリングデバッガDEBUG，スクリーンエディタSCREDなどの各種ツールが収められている。さらには漢字対応版スクリーンエディタKμMACSも付属しているため，プログラム記述がより簡単に行える。このパッケージの内容に関しては，本文89ページにも紹介してあるので，そちらも参考にしてほしい。

X68000用　5″2HD版　58,000円
マイクロウェア・ジャパン　☎0474(22)1747

★プログラマーズ・ツール・キット

このOS-9/X68000用の「プログラマーズ・ツール・キット」は，主にドライバ開発を支援するためのパッケージで，システムステートで動作するシンボリックデバッガSYSDBGや，ROMデバッガ ROMDBGの２つのデバッガと，SFC系のドライバサンプルなど２つのソースコードがパッケージされているほか，マクロアセンブラR68，リンカL68，漢字フルスクリーンエディタKμMACSなど，「C&プロフェッショナルパッケージ」に含まれているものと同じアセンブラ関係のツールもセットとなっている。また，このキットに付属のシステムインプリメンテーション・マニュアルには，ドライバの構造，作成方法などが詳しく紹介されている。

X68000用　5″2HD版　価格未定
マイクロウェア・ジャパン　☎0474(22)1747

MZ-700用SPACE HARRIERをX1で

Shimono Toshinori 下野 俊典
Sunagawa Namimichi 砂川 波路

なんと，あのMZ-700版のSPACE HARRIERがX1に移植されました。追加訂正はわずか3Kバイト。驚異の高速3Dキャラクタグラフィック画面を存分に堪能してください。これで「X1にも不可能はない」，かな?

MZ-700だけじゃない

この作品は1988年度GAME OF THE YEAR賛同作品です。最近ではSPACE HARRIERは数多くの機種に移植されています。当然，X1にも電波新聞社からちゃんとした製品が市販されているのですが，8ビットマシンでこのようなとてつもないゲームを実現するには辛いものがあります。これはPC-8801版やFM77AV版でも同様で，PC Engineやメガドライブといった高性能ゲームマシンでも苦しいSPACE HARRIERをMZ-700では3.5MHzのZ80で実現しているのです。

画面表示はすべてチェッカーマークで埋めつくされているため，細かな部分はわかりにくいのですが，色数が豊富なため表現力はなかなかのものです。近くではよくわかりませんが，ちょっと離れて見ると実によく雰囲気が出ています。処理速度にもまだまだ余裕があり，X68000版並みの浴びせるような高速連射も可能と，本当によくできています。まったく作者の古籏氏には脱帽ものです。

これをMZ-700ユーザーだけでなくX1ユーザーにも遊んでもらって，GAME OF THE YEARには700版として投票してもらおう，というのがこのプログラムの狙いです。

入力方法

まず，Oh!X 1988年10月号を用意してください。古籏氏によるMZ-700版のSPACE HARRIERが掲載されていますので，S-OS“SWORD”を使って，そのリストを2000Hずらして打ち込みます(すなわち1200Hからのリストを3200Hから入力するわけです)。プログラムの実行時にも“SWORD”が必要なのでまだ入力してない人はこちらを先に入力してください。

プログラムはマシン語モニタのMコマンドかマシン語入力ツールMACINTO-Cを使用して入力してください。打ち込んだプログラムはS-OS上から，

#S HARRIER700:3200:DB7F:1FFA

のようにセーブしておいてください。もちろん，お近くにすでにリストを入力したMZ-700ユーザーがいれば，S-OSを介してもらってきてもかまいません。

これを打ち込んでしまえば，あとはたった3Kバイトです。とりあえず，リスト1～4のプログラムを同じようにして打ち込み，それぞれセーブします。

まず，アドレスをずらしたMZ-700版にパッチあてを行います。セーブしておいたMZ-700版のSPACE HARRIERとリスト1のパッチあてプログラムをメモリ上に読み込んでください。

#JE000

で瞬間的にパッチあてが完了しますので，前もって入力しておいたリスト2～4をメモリに読み込み，それらをまとめて，

#S SPACE HARRIER:3200:E7FF:DB80

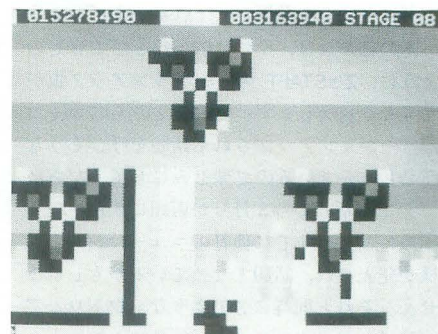
のようにセーブします。

プログラムを実行するには，

#JDB80

とします(実行アドレスが先頭アドレスとは異なりますので注意してください)。PCG定義に数秒かかりますが，タイトルが表示されデモが始まります。

操作方法

スペースキーかジョイスティックのボタンを押すことでゲームが開始します。キーボードではテンキーでキャラクターの移動，スペースキーで弾の発射を行います。X1ではキーの同時入力ができませんのでなるべくジョイスティックを使用したほうがよいでしょう。

ジョイスティックの場合，トリガー1はごくふつうの弾発射ボタンですがトリガー2では自動連射が可能です。また，リバースモード(スティックを上にあげるとキャラクターが下にさがる)とノーマルモード(スティックを上にあげるとキャラクターが上にあがる)の選択はゲーム中に“R”,“N”キーを押すことにより行われます(大文字のみ有効なので注意)。

X1turboまたはFM音源ボードを搭載しているX1ではシフトキー+“1”により，BGM(メインテーマのみ)を演奏することができます。演奏停止の際はシフトキー+“2”を押してください。

そのほか，「ありがちな機能」についても

図1 メモリマップ

サポートしてありますのでいろいろと探してみてください。

プログラムについて

PCG，BIOSモドキ，パッチ用プログラムがそれぞれ1Kバイトずつの3Kバイト。はっきりいって中身はグチャグチャです（特にBIOSモドキ）。途中まではアセンブラを使ってパッチをあてていましたが，後半はハンドアセンブルです。おかげでかなり汚くなってしまいました。

サウンドはPSGの1声のみでMZ-700以上のことはなにもやっていません。今回の移植で最後まで手がかかったのが，このサウンドの部分です。実はサウンドは苦手です。いろいろ苦労して，このような感じに落ち着いたのですが，デキはどうでしょうか。

MZ-700はキャラクタ単位に文字色と背景色を指定できるのですが，このようなことはX1ではできません。そこで使用する色をすべてPCGに定義しておきました。ちなみに，PCG定義には『試験に出るX1』から3倍速PCG定義ルーチンを使わさせてもらっています。

X1独自の機能をまったくといっていいほど使っていないので不満を感じる方もいるかもしれませんが，MZ-700版をX1で再現してみたかったのですからこれでいいのです。それから，スピード的にはMZ-700版よりも少し速くなっていますので，ゲームは難しくなったかもしれません。連射で弾を撃ちっぱなしにすれば少しは加減できます。

X1のPCGでできることならば，MZ-2500ででも当然同じようなことができるわけですが，誰か挑戦してみませんか？

Profile

◇下野さんは山口県にお住まいの21歳，大学3年生です。中学2年のときPC-6001を触って以来，主にグラフィックプログラムを作っていたとか。X1turboIIIは最近購入したばかりです。

◇砂川さんは山口県にお住まいの20歳。高校2年のころからPC-8801を使い，プログラムの解析やシステムプログラムなどを作っていたそうです。

リスト1　パッチあてプログラム

```
E000 21 00 E1 5E 23 56 23 7A : 76
E008 A3 3C C8 46 23 7E 23 12 : C3
E010 13 10 FA 18 EE 00 00 00 : 23
E018 00 00 00 00 00 00 00 00 : 00
E020 00 00 00 00 00 00 00 00 : 00
E028 00 00 00 00 00 00 00 00 : 00
E030 00 00 00 00 00 00 00 00 : 00
E038 00 00 00 00 00 00 00 00 : 00
E040 00 00 00 00 00 00 00 00 : 00
E048 00 00 00 00 00 00 00 00 : 00
E050 00 00 00 00 00 00 00 00 : 00
E058 00 00 00 00 00 00 00 00 : 00
E060 00 00 00 00 00 00 00 00 : 00
E068 00 00 00 00 00 00 00 00 : 00
E070 00 00 00 00 00 00 00 00 : 00
E078 00 00 00 00 00 00 00 00 : 00
-----------------------------------
SUM: D7 4C A3 BC 34 D4 46 8C 95D8

E080 00 00 00 00 00 00 00 00 : 00
E088 00 00 00 00 00 00 00 00 : 00
E090 00 00 00 00 00 00 00 00 : 00
E098 00 00 00 00 00 00 00 00 : 00
E0A0 00 00 00 00 00 00 00 00 : 00
E0A8 00 00 00 00 00 00 00 00 : 00
E0B0 00 00 00 00 00 00 00 00 : 00
E0B8 00 00 00 00 00 00 00 00 : 00
E0C0 00 00 00 00 00 00 00 00 : 00
E0C8 00 00 00 00 00 00 00 00 : 00
E0D0 00 00 00 00 00 00 00 00 : 00
E0D8 00 00 00 00 00 00 00 00 : 00
E0E0 00 00 00 00 00 00 00 00 : 00
E0E8 00 00 00 00 00 00 00 00 : 00
E0F0 00 00 00 00 00 00 00 00 : 00
E0F8 00 00 00 00 00 00 00 00 : 00
-----------------------------------
SUM: 00 00 00 00 00 00 00 00 0000

E100 00 32 02 00 00 0D 32 06 : 79
E108 CD B0 03 00 00 00 17 32 : C9
E110 02 00 00 31 32 03 00 00 : 68
E118 00 23 36 01 00 25 36 04 : B9
E120 00 3E FF 00 2C 36 0F 00 : AE
E128 00 00 00 00 00 00 00 00 : 00
E130 00 00 00 00 00 00 FA 36 : 30
E138 0D CD 60 01 00 00 00 00 : 3B
E140 00 00 00 01 00 00 12 37 : 4A
E148 08 CD 81 01 00 00 00 00 : 57
E150 00 1B 37 03 00 00 00 32 : 87
E158 37 08 00 00 00 00 00 00 : 3F
E160 00 00 3B 37 04 00 00 00 : 76
E168 00 49 37 06 91 01 00 00 : 18
E170 00 00 50 37 10 00 00 00 : 97
E178 00 00 00 00 00 00 00 00 : 00
-----------------------------------
SUM: 1B 49 14 AC 03 6C 9A DB 83F0

E180 00 00 00 00 00 62 37 01 : 9A
E188 33 71 37 03 CD 51 01 7C : 79
E190 38 2C 28 29 2A 28 29 3B : 6B
E198 29 2B 2C 29 2B 3C 3D 3E : 8B
E1A0 3C 3D 3E 3C 3D 3E 3C 3F : E9
E1A8 2F 2E 39 2D 38 39 3A 2E : 9C
E1B0 39 2D 2E 38 38 3B 38 38 : AF
E1B8 3B 38 38 3B 3B 38 BE 38 : 4F
E1C0 06 3A 12 00 EE 30 00 C5 : 35
E1C8 38 02 05 00 CD 38 02 05 : 4B
E1D0 00 D5 38 02 05 00 E8 38 : 34
E1D8 02 05 00 F7 38 02 05 00 : 3D
E1E0 28 3A 05 3E 0D C3 52 02 : C9
E1E8 3E 3A 01 00 40 3A 06 3A : 33
E1F0 12 00 00 FE 26 87 3B 08 : 00
E1F8 00 00 00 00 00 00 00 00 : 00
-----------------------------------
SUM: 2B 22 BD 66 75 EF 8C 19 3822

E200 90 3B 04 00 00 00 00 A7 : 76
E208 3B 01 00 A9 3B 12 00 3E : 70
E210 FF 00 00 00 00 00 00 00 : FF
E218 3A 12 00 00 00 FE 27 C2 : 33
E220 D7 3B 0B 3A 12 00 FE 21 : 88
E228 C0 00 00 00 00 00 EF 3D : EC
E230 01 00 F1 3D 06 3A 12 00 : 81
E238 00 FE 22 15 A0 02 FF FF : D5
E240 A9 A2 02 18 19 AE A2 13 : E1
E248 3A 6A CC C6 08 32 8E C7 : C5
E250 C9 00 00 00 00 00 00 00 : C9
E258 00 00 00 CC A2 04 C0 CD : FF
E260 BB 01 D4 A2 04 C0 CD C3 : 86
E268 01 6D AA 06 CD 80 00 00 : 6B
E270 00 00 2A B0 03 C3 DC 00 : 7C
E278 2E B0 2A E1 08 E9 07 EF : D0
-----------------------------------
SUM: 32 B1 C2 18 92 1C C5 5D 7CCF

E280 0E 4D 0D DA 0B 2F 0B F7 : 7E
E288 09 62 08 E9 07 17 0E 8E : 16
E290 0C 2F 0B 8F 0A 68 09 62 : B2
E298 08 E9 07 17 0E 8E 0C 2F : E6
E2A0 0B 8F 0A 68 09 CE B0 05 : 98
E2A8 00 CD C7 00 00 FB B1 03 : 43
E2B0 C3 C7 00 0F B2 03 C3 C7 : D8
E2B8 00 37 B2 03 C3 C7 00 7C : F2
E2C0 B2 01 34 82 B2 01 35 0F : 60
E2C8 B4 03 CD C0 00 3B B4 03 : 36
E2D0 CD 33 01 4B B4 03 CD 33 : 03
E2D8 01 56 B4 01 00 70 B6 02 : 34
E2E0 E0 00 8E B6 02 30 32 BD : 45
E2E8 B6 02 30 34 E2 B6 02 30 : E6
E2F0 34 18 B7 02 30 31 5D B7 : 7A
E2F8 02 30 32 7C B7 02 30 41 : 0A
-----------------------------------
SUM: F9 F8 07 D9 D9 97 7F 8D 945F

E300 84 B7 02 30 30 19 B9 02 : 71
E308 30 30 20 B9 02 30 38 15 : B8
E310 C0 02 FF FF 81 C0 07 C9 : D1
E318 00 00 00 00 00 00 88 C8 : 50
E320 01 00 8A C8 04 CD A0 03 : C7
E328 00 8F C8 01 57 7F CA 01 : F9
E330 30 88 CA 01 30 8A CA 01 : 08
E338 C0 8D CA 03 CD CB 01 91 : 44
E340 CA 03 CD CB 01 DE CF 06 : 19
E348 CD 80 00 00 00 00 EA CF : 06
E350 01 40 F5 CF 01 48 1B D0 : 39
E358 04 C3 50 02 01 20 D0 02 : 0C
E360 30 16 23 D0 1C D5 16 28 : 68
E368 CB A0 3E 27 ED 79 CB E0 : E1
E370 7E 23 ED 79 03 15 20 F0 : 2F
E378 11 18 00 19 D1 15 20 E5 : 2D
-----------------------------------
SUM: 8B 04 67 DA EB 68 7A C2 4596

E380 C9 BB D2 03 00 00 00 BF : 18
E388 D2 0C 00 00 00 00 CD D4 : 7F
E390 01 CB 77 CA 1D B3 CC D2 : 7B
E398 04 F6 0F 00 00 73 D3 02 : 51
E3A0 3A 12 76 D3 17 FE 1B C0 : 85
E3A8 CD 1B 00 B7 20 FA CD 1B : A1
E3B0 00 B7 28 FA CD 1B 00 FE : BF
E3B8 1B 28 F9 C9 77 D6 01 00 : 53
E3C0 8E D6 01 00 9A D6 04 C1 : 9A
E3C8 CD AB 01 A2 D6 04 C0 CD : 82
E3D0 B3 01 FF FF FF FF FF FF : AE
E3D8 FF FF FF FF FF FF FF FF : F8
E3E0 FF FF FF FF FF FF FF FF : F8
E3E8 FF FF FF FF FF FF FF FF : F8
E3F0 FF FF FF FF FF FF FF FF : F8
E3F8 FF FF FF FF FF FF FF FF : F8
-----------------------------------
SUM: CB 11 EB B6 02 E3 13 C8 778A
```

リスト2　START

```
DB80 01 00 10 AF ED 79 04 ED : 17
DB88 79 04 ED 79 04 3E 50 CD : 42
DB90 30 20 CD 00 E0 3E 28 CD : 30
DB98 30 20 3E E4 CD BE DB 3E : 16
DBA0 10 F3 CD C7 DB 21 00 32 : C5
DBA8 11 00 12 01 80 A9 ED B0 : EA
DBB0 21 00 E4 11 00 00 01 00 : 17
DBB8 04 ED B0 C3 00 12 FB CD : 3E
DBC0 C7 DB CD D4 DB F3 C9 C5 : 9F
DBC8 F5 CD D4 DB F1 01 00 19 : 7C
DBD0 ED 79 C1 C9 01 01 1A ED : F9
DBD8 78 E6 40 20 FA C9 3D 00 : BE
DBE0 3D 00 3D 00 3D 00 3D 00 : F4
DBE8 3D 00 3D 00 3D 00 3D 00 : F4
DBF0 3D 00 3D 00 3D 00 3D 00 : F4
DBF8 3D 00 3D 00 3D 00 3D 00 : F4
-----------------------------------
SUM: 35 2B 11 40 B4 4D 54 3F CB58
```

リスト3　BIOS

```
E400 C3 00 12 00 00 00 00 00 : D5
E408 00 00 00 00 00 00 00 00 : 00
E410 56 03 00 00 00 C3 3F 00 : 5B
E418 C3 00 02 C3 40 00 ED 79 : 2E
E420 CB A0 3E 07 ED 79 CB E0 : C1
E428 03 C9 00 00 00 00 00 00 : CC
E430 00 00 00 00 00 00 00 00 : 00
E438 00 00 00 00 00 00 C9 00 : C9
E440 00 3A 12 00 B7 C0 F3 C5 : 7B
E448 01 00 1C 3E 0E ED 79 05 : D4
E450 ED 78 2F C1 FB C9 00 00 : 19
E458 00 00 00 00 00 00 00 00 : 00
```

```
E460 00 00 00 00 00 00 31 00 : 31
E468 00 C3 00 12 00 00 F3 C5 : 8D
E470 01 00 1C 3E 0E ED 79 05 : D4
E478 ED 78 2F C1 FB C9 00 00 : 19
---------------------------------
SUM: 86 59 FA DA F6 68 C9 ED 65DD

E480 3A 12 00 B7 C2 A0 00 F3 : 58
E488 C5 01 00 1C 3E 0E ED 79 : 94
E490 05 ED 78 E6 0F 4F 3A 13 : FB
E498 00 CD 30 02 79 C1 FB C9 : FD
E4A0 D6 31 DA B3 00 FE 09 D2 : 6D
E4A8 B3 00 E5 26 00 C6 B6 6F : A9
E4B0 7E E1 C9 3E FF C9 F9 FD : 24
E4B8 F5 FB FF F7 FA FE F6 00 : D4
E4C0 CB 3C CB 1D 10 FA C9 C5 : 87
E4C8 F5 01 00 1C AF ED 79 05 : 2C
E4D0 ED 59 04 3C ED 79 05 ED : DE
E4D8 51 F1 C1 C9 F1 FB ED 4D : F2
E4E0 3E 38 26 07 CD 2A 01 3E : D9
E4E8 0F 26 08 CD 2A 01 24 CD : 26
E4F0 2A 01 24 CD 2A 01 01 A0 : E8
E4F8 1F 3E 03 ED 79 03 ED 79 : 2F
---------------------------------
SUM: 94 FE 14 95 B8 D3 17 AE E468

E500 03 ED 79 03 ED 79 21 00 : F3
E508 90 22 5E 00 F3 01 A0 1F : C3
E510 3E 27 ED 79 3E 03 ED 79 : 72
E518 3E 58 ED 79 0E A3 3E C7 : B2
E520 ED 79 3E 20 ED 79 FB C3 : E8
E528 12 90 01 00 1C ED 61 05 : 12
E530 ED 79 C9 FD 46 0F 4F AF : 7F
E538 81 10 FD FD 77 08 C9 ED : C0
E540 78 77 23 03 15 20 F8 C9 : 0B
E548 00 00 00 00 00 00 00 00 : 00
E550 00 C5 44 4D ED 79 DD 23 : BC
E558 C1 C9 00 00 00 00 00 00 : 8A
E560 01 00 30 21 E8 03 AF ED : D9
E568 79 CB A0 3E 27 ED 79 CB : 7A
E570 E0 03 2B 7C B5 20 EF C9 : 17
E578 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 0F F3 18 3A B8 46 4C 30 C1C5

E580 00 01 C0 33 21 28 00 AF : EC
E588 ED 79 03 2B 7C B5 20 F7 : DC
E590 C9 01 00 20 3E 2D 26 03 : 7E
E598 ED 79 03 25 20 FA 0E 0F : C5
E5A0 3E 2A 26 05 ED 79 03 25 : 21
E5A8 20 FA C9 C5 01 7E 31 ED : 45
E5B0 79 C1 C9 C5 01 7F 31 ED : 66
E5B8 79 C1 C9 C5 01 25 30 ED : 0B
E5C0 79 C1 C9 C5 01 26 30 ED : 0C
E5C8 79 C1 C9 ED 6F C5 42 4B : B1
E5D0 ED 79 C1 C9 3A 12 00 FE : 3A
E5D8 20 C8 F3 C5 01 00 1C 3E : FB
E5E0 0E ED 79 05 ED 78 C1 FB : 9A
E5E8 C9 FF 00 FF 00 FF 00 FF : C5
E5F0 CD 3F 02 32 13 00 C9 FF : 1B
E5F8 00 FF 00 FF 00 FF 00 FF : FC
---------------------------------
SUM: 96 87 08 6C 96 12 01 10 0DFF

E600 E5 F5 D5 C5 2A 71 11 7D : 9D
E608 6C 26 00 44 4D 29 29 09 : 7E
E610 29 29 29 4F 06 00 09 01 : DA
E618 00 30 09 44 4D 1A 13 FE : F5
E620 0D 28 07 C6 90 ED 79 03 : FB
E628 18 F3 C1 D1 F1 E1 C9 00 : 38
E630 B7 C8 CB 19 F5 CB 09 F1 : 1D
E638 CB 11 CB 11 C9 00 00 C5 : 46
E640 01 01 1A ED 78 E6 20 20 : A7
E648 FA 01 00 19 ED 78 C1 C9 : 03
E650 3E 18 21 CC C1 01 28 30 : 5D
E658 16 00 08 1E 7F 7E 23 A3 : FF
E660 ED 79 03 7E 23 A3 ED 79 : 13
E668 03 7E 23 A3 ED 79 03 7E : 2E
E670 23 A3 ED 79 03 7E 23 A3 : 73
E678 ED 79 03 7E 23 A3 ED 79 : 13
---------------------------------
SUM: 70 95 BE 65 E4 67 CD 0D 682D

E680 03 7E 23 A3 ED 79 03 7E : 2E
E688 23 A3 ED 79 03 7E 23 A3 : 73
E690 ED 79 03 7E 23 A3 ED 79 : 13
E698 03 7E 23 A3 ED 79 03 7E : 2E
E6A0 23 A3 ED 79 03 7E 23 A3 : 73
E6A8 ED 79 03 7E 23 A3 ED 79 : 13
E6B0 03 7E 23 A3 ED 79 03 7E : 2E
E6B8 23 A3 ED 79 03 7E 23 A3 : 73
E6C0 ED 79 03 7E 23 A3 ED 79 : 13
E6C8 03 7E 23 A3 ED 79 03 7E : 2E
E6D0 23 A3 ED 79 03 7E 23 A3 : 73
E6D8 ED 79 03 7E 23 A3 ED 79 : 13
E6E0 03 7E 23 A3 ED 79 03 7E : 2E
E6E8 23 A3 ED 79 03 7E 23 A3 : 73
E6F0 ED 79 03 7E 23 A3 ED 79 : 13
E6F8 03 7E 23 A3 ED 79 03 7E : 2E
---------------------------------
SUM: 62 80 82 A5 4C 7B 62 80 990A

E700 23 A3 ED 79 03 7E 23 A3 : 73
E708 ED 79 03 7E 23 A3 ED 79 : 13
E710 03 7E 23 A3 ED 79 03 7E : 2E
E718 23 A3 ED 79 03 7E 23 A3 : 73
E720 ED 79 03 7E 23 A3 ED 79 : 13
E728 03 7E 23 A3 ED 79 03 7E : 2E
E730 23 A3 ED 79 03 7E 23 A3 : 73
E738 ED 79 03 7E 23 A3 ED 79 : 13
E740 03 7E 23 A3 ED 79 03 7E : 2E
E748 23 A3 ED 79 03 1E 18 19 : 7E
E750 08 3D C2 5A 02 C9 F3 F5 : 14
E758 CD 3F 02 CD 3F 02 32 12 : 60
E760 00 FE 52 CA 6C 03 FE 4E : D5
E768 C2 6F 03 AF 32 13 00 F1 : 19
E770 FB ED 4D 00 00 00 00 00 : 35
E778 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: EE 47 8C E7 1B CD 74 2D 46A2

E780 FB CD 89 03 CD 96 03 F3 : AD
E788 C9 C5 F5 CD 96 03 F1 01 : DB
E790 00 19 ED 79 C1 C9 01 01 : 0B
E798 1A ED 78 E6 40 20 FA C9 : 88
E7A0 3E E6 CD 80 03 CD 3F 02 : 82
E7A8 F5 CD 3F 02 F1 C9 00 00 : BD
E7B0 01 A0 1F 3E 47 ED 79 3E : E9
E7B8 5A 5F ED 79 ED 78 BB CA : 09
E7C0 E6 03 01 04 07 3E 47 ED : 67
E7C8 79 7B ED 79 ED 78 BB C2 : 3C
E7D0 ED 03 21 04 07 3E 07 22 : 83
E7D8 F7 00 22 0E 01 32 1D 01 : 78
E7E0 3E F3 32 00 96 C9 21 A0 : 83
E7E8 1F 3E A3 18 EA 3E C9 32 : 3B
E7F0 00 96 C9 00 00 00 00 00 : 5F
E7F8 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 0C 92 CA 0F 08 AA 72 6C 348E
```

リスト4 PCG

```
E000 0E 77 1E 08 16 08 7A 3D : 80
E008 CD 66 E0 D5 7B 3D EB CD : 58
E010 66 E0 CD 73 E0 C5 59 21 : A5
E018 FA E0 01 D0 07 16 30 CD : C5
E020 12 E1 C1 D1 0D 15 20 DE : A5
E028 79 D6 08 4F 1D 20 D5 CD : 85
E030 92 E0 21 40 E2 01 D0 07 : 8D
E038 16 30 1E 18 CD 12 E1 21 : 5D
E040 58 E2 06 08 1E 28 CD 50 : AB
E048 E0 21 18 E3 06 08 1E 38 : 60
E050 C5 01 D0 07 D5 16 30 E5 : 9D
E058 CD 12 E1 E1 11 18 00 19 : E3
E060 D1 1C C1 10 EB C9 C5 06 : 3D
E068 00 4F 87 81 4F 21 E2 E0 : 89
E070 09 C1 C9 C5 3E 03 01 FA : 94
E078 E0 08 1A 0F B6 13 23 E5 : E2
---------------------------------
SUM: F2 AE CE D0 89 C6 7A 16 A392

E080 26 04 02 03 0F 02 03 0F : 52
E088 25 20 F7 E1 08 3D 20 E9 : 6B
E090 C1 C9 06 40 1E 20 16 90 : B4
E098 CD AA E0 06 08 1E 30 16 : C9
E0A0 D8 CD AA E0 06 01 1E 3D : 91
E0A8 16 DC D5 C5 01 D0 07 16 : 7A
E0B0 30 21 FA E0 CD CD E1 21 : C7
E0B8 FA E0 11 02 E1 01 08 00 : D7
E0C0 ED B0 21 FA E0 01 08 00 : A1
E0C8 ED B0 C1 D1 D5 C5 7B 82 : C6
E0D0 5F 01 D0 07 16 30 21 FA : 98
E0D8 E0 CD 12 E1 C1 D1 1C 10 : 5E
E0E0 C9 C9 00 00 00 AA 00 00 : 3C
E0E8 00 AA 00 AA AA 00 00 00 : FE
E0F0 AA AA 00 AA 00 AA AA AA : FC
E0F8 AA AA 00 00 00 00 00 00 : 54
---------------------------------
SUM: 27 36 2D B8 28 37 E1 48 F750

E100 00 00 00 00 00 00 00 00 : 00
E108 00 00 00 00 00 00 00 00 : 00
E110 00 00 ED 43 B3 E1 CD 1D : AE
E118 E1 CD 57 E1 C9 E5 D5 01 : 6A
E120 D0 1F AF ED 79 ED 4B B3 : EF
E128 E1 21 00 38 3E 00 CD 4D : 92
E130 E1 ED 4B B3 E1 21 00 28 : F6
E138 D1 D5 3E 20 CD 4D E1 ED : EC
E140 4B B3 E1 21 00 30 D1 7B : 7C
E148 CD 4D E1 E1 C9 09 44 4D : 3F
E150 ED 79 03 15 20 FA C9 CD : 2E
E158 8F E1 06 16 0E 00 16 17 : C7
E160 1E 18 3E 08 08 D9 F3 01 : 51
E168 01 1A ED 78 F2 6A E1 ED : AA
E170 78 FA 6F E1 D9 08 ED A3 : 33
E178 42 ED A3 43 ED A3 06 16 : C1
---------------------------------
SUM: B1 42 84 ED 98 42 56 86 BD9C

E180 08 3E 0B 3D C2 83 E1 08 : BC
E188 0C 3D C2 76 E1 FB C9 D9 : FF
E190 06 08 D9 11 B5 E1 E5 01 : 74
E198 03 00 ED A0 E2 A7 E1 79 : 73
E1A0 0E 07 09 4F C3 9A E1 E1 : 8C
E1A8 23 D9 05 D9 C2 96 E1 21 : 34
E1B0 B5 E1 C9 00 00 00 00 00 : 5F
E1B8 00 00 00 00 00 00 00 00 : 00
E1C0 00 00 00 00 00 00 00 00 : 00
E1C8 00 00 00 00 00 ED 43 3B : 6B
E1D0 E2 CD DB E1 01 00 14 CD : 4D
E1D8 15 E2 C9 E5 D5 01 D0 1F : 6A
E1E0 AF ED 79 ED 4B 3B E2 21 : 8B
E1E8 00 38 3E 00 CD 0B E2 ED : 1D
E1F0 4B 3B E2 21 00 28 D1 D5 : 57
E1F8 3E 00 CD 0B E2 ED 4B 3B : 6B
---------------------------------
SUM: 32 53 74 6B 8F 7F 39 A2 8E63

E200 E2 21 00 30 D1 7B CD 0B : 57
E208 E2 E1 C9 09 44 4D ED 79 : 8C
E210 03 15 20 FA C9 1E 08 D9 : FA
E218 F3 01 01 1A ED 78 F2 1C : 82
E220 E2 ED 78 FA 21 E2 D9 ED : 0A
E228 78 77 23 03 00 23 2B 3E : A1
E230 0D 3D C2 31 E2 1D C2 27 : 25
E238 E2 FB C9 00 00 00 00 00 : A6
E240 00 38 38 10 7C 10 28 44 : 78
E248 00 38 38 10 7C 10 28 44 : 78
E250 00 38 38 10 7C 10 28 44 : 78
E258 01 03 07 0F 1F 3F 7F FF : F6
E260 00 00 00 00 00 00 00 00 : 00
E268 01 03 07 0F 1F 3F 7F FF : F6
E270 FF FF FF FF FF FF FF FF : F8
E278 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 04 61 C5 C8 7F 2D EF 94 4419

E280 FF FF FF FF FF FF FF FF : F8
E288 FF FE FC F8 F0 E0 C0 80 : 01
E290 00 00 00 00 00 00 00 00 : 00
E298 FF FE FC F8 F0 E0 C0 80 : 01
E2A0 80 C0 E0 F0 F8 FC FE FF : 01
E2A8 00 00 00 00 00 00 00 00 : 00
E2B0 80 C0 E0 F0 F8 FC FE FF : 01
E2B8 FF 7F 3F 1F 0F 07 03 01 : F6
E2C0 00 00 00 00 00 00 00 00 : 00
E2C8 FF 7F 3F 1F 0F 07 03 01 : F6
E2D0 FF 80 80 80 80 80 80 80 : 7F
E2D8 00 00 00 00 00 00 00 00 : 00
E2E0 FF 80 80 80 80 80 80 80 : 7F
E2E8 FF 01 01 01 01 01 01 01 : 06
E2F0 00 00 00 00 00 00 00 00 : 00
E2F8 FF 01 01 01 01 01 01 01 : 06
---------------------------------
SUM: F7 7B 37 0F EF C7 83 01 5CB7

E300 FF 00 00 00 00 00 00 00 : FF
E308 00 00 00 00 00 00 00 00 : 00
E310 FF 00 00 00 00 00 00 00 : FF
E318 FF FF 00 00 00 00 00 00 : FE
E320 00 00 00 00 00 00 00 00 : 00
E328 FF FF 00 00 00 00 00 00 : FE
E330 01 01 01 01 01 01 01 01 : 08
E338 00 00 00 00 00 00 00 00 : 00
E340 01 01 01 01 01 01 01 01 : 08
E348 80 80 80 80 80 80 80 80 : 00
E350 00 00 00 00 00 00 00 00 : 00
E358 80 80 80 80 80 80 80 80 : 00
E360 00 00 00 00 00 00 00 00 : 00
E368 00 00 00 00 00 00 00 00 : 00
E370 00 00 00 00 00 00 00 00 : 00
E378 FF 80 80 80 80 80 80 80 : 7F
---------------------------------
SUM: FD 80 82 82 82 82 82 82 5EE4

E380 00 00 00 00 00 00 00 00 : 00
E388 00 00 00 00 00 00 00 00 : 00
E390 FF FF 00 00 00 00 00 00 : FE
E398 00 00 00 00 00 00 00 00 : 00
E3A0 00 00 00 00 00 00 00 00 : 00
E3A8 01 01 01 01 01 01 01 01 : 08
E3B0 00 00 00 00 00 00 00 00 : 00
E3B8 00 00 00 00 00 00 00 00 : 00
E3C0 FF 01 01 01 01 01 01 01 : 06
E3C8 00 00 00 00 00 00 00 00 : 00
E3D0 00 00 00 00 00 00 00 00 : 00
E3D8 00 00 00 00 00 00 00 00 : 00
E3E0 00 00 00 00 00 00 00 00 : 00
E3E8 00 00 00 00 00 00 00 00 : 00
E3F0 00 00 00 00 00 00 00 00 : 00
E3F8 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: FF 01 02 02 02 02 02 02 08C5
```

特集

BASIC“おもちゃ箱”

CONTENTS

OTAKU

What
When
Where

あがり

ふりだし

BLOCK

BASIC

Animation

3D

Gravity

どんなコンピュータにも(ファミコンからメインフレームまで),ハードウェアとソフトウェアがあります。しかし,すべてのユーザーに「自分でソフトを作る環境」が与えられているのはパソコンだけの大きな魅力です。もちろんプログラミング環境は機種によっても異なりますが,Oh!Xをお読みの皆さんが共通して持っている環境,それがBASICです。おそらくパソコンでプログラムを組もうとする人が最初に使うのもBASICでしょう。

確かに,BASICはプログラミング言語としては完全なものではないかもしれません。ある程度プログラミング経験のある人なら不満のタネはいくらでもあるでしょう。それでも,何かを作ってみたいと思い立ったとき,とりあえず試してみようというときに,BASICに取って代わるものはないといってよいくらいです。アルゴリズムの検討など,プログラミング自体のシミュレータとしても有効でしょう。BASICで思考することは,ちょうど子供がおもちゃ箱をひっくり返して,自ら「遊び」を創造するのと似ています。その意味ではBASICは万能です。

また,ユーザーのわがままをすべて許してくれる寛容さも大きな特長です。たとえ,プログラムの大半がマシン語ルーチンに占領されても BASIC はメインルーチンとしての立場をまっとうしてくれます。マシン語プログラムを使うための道具にしてしまっても文句ひとつ言いません。PCGを定義するだけでも,サウンドデータを流すだけでも立派なプログラムとなります。そう,BASICプログラミングにこうでなくては,といった作法はないのです。

今回の特集では,このBASICを使ったプログラミングのさまざまな楽しみ方を見てほしいと思います。ともかく,BASICでプログラミングの遊び場を作ってみましょう。

BASIC文学書き下ろし特別作品

世界の終りとベーシック・ワンダーランド

よーするに「がんばれ！カズシゲ君MZ-2500版」

PCGのカズシゲ君が見せる華麗なプレイにもそれなりのドラマが隠されていたのです。果たしてBASICは文学になるか？　というわけで村上春樹さんごめんなさい。

Ogikubo　Kei
荻窪　圭

1　世界の終り ―ジュニアの時代―

マスコミの端っこのほうでは，ジュニア(2世)の時代だとか，いわれているそうだ。何やら政財界とかいう怪しげな世界では，親父さんの跡を継いだ連中が大きな顔をしているそうで，ああ，終戦から40年以上も経っているのだもんなあと思わせるものがある。聞くところによると，2世代議士には保守的な人が多いそうだが，まあ，だいたい，始めから財産を守りに入っているのだから，持たぬ者より保守的なのは当たり前で，歴史が示すところに従うと，そういった社会はロクなほうに向かった例しがない。昔と今を一緒にしてはいけない――などという人が必ずいるが，人間自体がたいして進歩していないのだから，そうそう飛び出していけるわけもない。いずれにしても，ひとつの世界が名を持つ者に左右されるというのは私は好ましく思っていない。

一方スポーツ界では，こちらもジュニアと呼ばれる人が相撲から水泳から野球までいろいろ出現しているが，いくら親が金持ちで名があっても，実力は基本的に裸一貫であるからして，親の名というプレッシャーの下でのし上がっていくのは大変だろうといえる。血筋が云々とよくいわれるが，才能なんて生まれついて備わっているものじゃない。環境と努力と運と偶然によって育まれていくものだ。自分の欲しかった才能が育まれなかったからといって，血を恨むのは筋違いである。恨むのなら，自分をそうは育ててくれなかった環境を恨みなさい（と，私は思う）。

その点，環境的には，親を見て子は育つものであるから，一流選手の子供は恵まれているといえないこともない。自然に身についてしまう事柄が多いから。ああ，私も音楽家の子供に生まれていたらもう少し音感もよかったろうし（いい音楽に囲まれて育つからね），大文学者の子供に生まれていたら群像新人賞くらいは取っていたかもしれない（有名人の子供だとそれだけでばかな大人が注目してくれるから）。

2　ベーシック・ワンダーランド ―テーブルってなに？―

今回はテーブルで迫ってみたい。ここでいうテーブルというのは，あらかじめ必要なデータが詰め込んである表のことである。あまりにも気楽に使われる用語なので，初心者の方は赤丸を付けておくように。

テーブルというのは，もちろん食事をしたり手紙を書いたりするときにも有効だが，プログラムにおいても実に便利である。

誰でも思いつくことだが，たとえば，RPGを作ろうと思ったとしよう。すると，モンスターの強さやらHPやら名前やらが必要となる。で，そういうときは，配列かなんかに表にして入れておくだろう。(1, n）が名前で，(2, n）がHPで，という具合。これも立派なテーブルである。

テーブル。BASICでいえば，配列にあらかじめ格納されたデータの集まりであるが，メモリを消費するという欠点にさえ目をつぶれば，いくらでも使い道はある。

たとえば，と，テーブルを駆使して，プログラムを作ってみた。

3　世界の終り ―期待のカズシゲ―

あの，ヤクルトに昨年入団した長島一茂もそういった2世選手といわれているもののひとりであることは間違いない。だいたい，彼の場合，ヤクルトに入ったことがそもそもの幸運であったといえよう。なおかつ，監督が関根さんだったというのが輪をかけて幸運だ。あの，関根監督はいい監督である。何もしていなさそうな好々爺という感じがいい。東の好々爺が関根監督ならば，西の好々爺はダイエーホークスの杉浦監督である。この2人には頑張ってもらいたいものだ。

話はそれたけど，そのようにして，長島一茂選手は育っていくのだが，2年目のキャンプ真っただ中のこの季節に際して，あれだけ注目されていてポシャっちゃったら可哀相だなあ，と，つい思ってしまう日本的人情派のあなたに送るのが，一茂君特訓ゲーム，「がんばれ！　カズシゲ君」である。いまこそここに掲載しよう。

まず，野球選手に大切なのは足腰の強さである。私のように20代半ばにして，寒くなると腰が痛むようでは情けない。

足腰を強くするには走り込め，などといわれているが，それも単調で退屈である。それより，足腰が鍛えられて，瞬発力がつくものがある。それは，ロッテの有藤監督が大好きな，ノックというやつである。

テーブルの効用

講談社はブルーバックスのコンピュータ用語事典によると，「主記憶装置ないし，他の記憶媒体中のデータの配列」である。

たとえば乱数表である。どこかの数学の本に載っている乱数表を配列に入れておけば，RND関数を使うのと同様の効果が，配列のある点から順に読み出していくだけで得られる。これは1次元の配列で得られるから簡単。暗号でも，換字式というのは，交わす同士が変換用のテーブルを使って暗号文を作ったり読んだりする。

2次元の配列となると，そこらじゅうに転がっている。たとえば，時間割り表。あれは曜日と時間からその科目を取り出すテーブルである。このように，あらかじめ蓄えられたデータをキーによって（時間割り表なら，曜日と時間がキーである）取り出せるように作られた表はすべてテーブルというのだ。

この考え方を応用して，たとえば，ある円周にそってくるくる回り続けるスプライトを表示したいとき，どうするか。毎回毎回三角関数を使って座標計算をするという労働をCPUに強いるか。同じ位置でくるくる回すのなら，あらかじめ，表示位置を計算して配列にいれておけばいいではないか。すると，配列を順にアクセスするだけでいいのでプログラムもすっきりするし，速くなる。これは便利である。

〈カズシゲ君の華麗な姿5態〉

4 ベーシック・ワンダーランド ―7つの配列―

というわけで，リスト1を見てもらいたいのは，それが「がんばれ！ カズシゲ君」のプログラムだからである。

このリストを試しに打ち込んでみるとよい。あっという間にエラーで止まってしまうだろう。では，どんなエラーでしょう。

まずこれはBASIC-M25でしか動かないプログラムであるが，なにもMZ-2500用だから，X1ユーザーはそのまま打ち込むとエラーになるよ，といった意味ではないから安心してほしい。

でもまあ，とりあえずリスト2のPCG定義プログラムは実行しておく必要がある。

リスト1に戻ろう。いきなり，7つもの配列が所狭しと並んでいる。しかも，ひとつは3次元，3つは2次元という荘厳さである。初心者がBASIC，いや，計算機上の言語に生まれて初めて取り組んだとき，最初につまずくのがこの配列という奴だ。

考え方は非常に簡単なのだけれども，ひとつの名前にいくつものデータが縦横無尽に放り込めるわけで，慣れないと，目に見えない表を管理するのはたいへんだろう。観念的にマクロな視点で理解してしまえば楽だけれども，(1,2) は何で，(2,5) はあれで，などと1つひとつを追ったりすると，瞬時にしてイヤになるからね。

5 世界の終り ―ミスターとジュニア―

ある日，ミスターがある多摩川かどっかの野球場へカズシゲを呼び出して，親子の戦いが始まった。星飛雄馬と星一徹か，はたまた中嶋常幸親子か，吉本隆明とばななか（んなことはないか）。

ミスターはホームベース付近にバットを持って立ち，カズシゲはサードベースの少し前に，背番号3を付けて構えた。

カズシゲは，前後の動きが苦手だから，左右にしか動けない。

ミスターはかけ声とともに，白球を打つ。

緑のジャングルに快音がこだまし，電撃のようにカズシゲは反応する。ダイヤモンドグラブ賞への道は果てしなく遠い。中日の立浪は人間じゃない，と，そのときカズシゲは思ったに違いない。

6 ベーシック・ワンダーランド ―すべてはテーブルにある―

リスト2によって定義されたPCGは，守るカズシゲの姿5態と，打つミスターの姿3態と，ボールである。

リスト1では，KAZ$の0～4にカズシゲの姿態が，SIG$の0～2にはミスターの勇姿が入る。

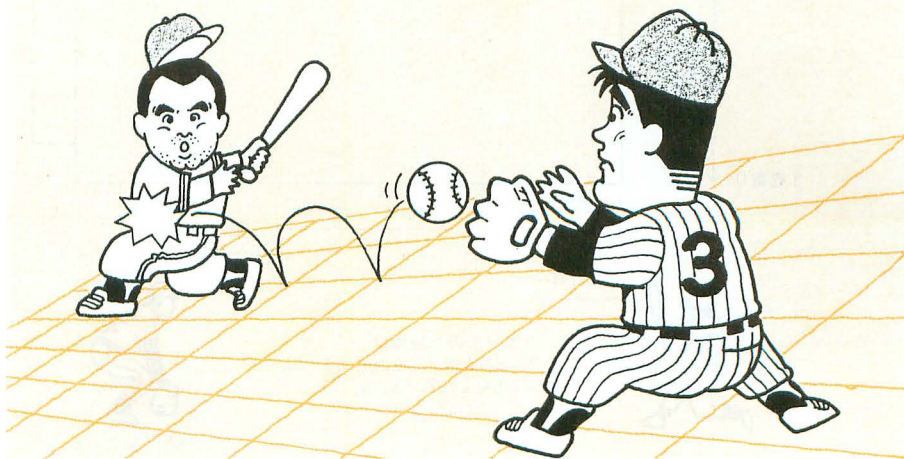
たとえば，カズシゲのPCGは縦2行，横3～4列であり，CHR$ (29) やら (32) やらはカーソル移動であるからして，まあそういったわけで，“PRINT KAZ$(2)”とすると，カズシゲが構えた姿が表示される。ちなみに，KAZ$ (0) は左にダイビングした姿，KAZ$ (1) は少し左にひねった姿，KAZ$ (3) は右に逆シングル，KAZ$ (4) は右にダイビングというわけである。

これで配列2つは解決した。で，残りであるが，ミスターがノックをして，カズシゲが受けるとなると，自ずと，必要なデータは決まってくる。まず，打球の位置である。それから，カズシゲの位置である。続いて，カズシゲの姿態である。

普通に考えると，打球は乱数で方向を決めて，Y軸方向にひとつずつ足しては消して書き，書いては消し，カズシゲもプレイヤーの入力があったら，その方向（この場合はX軸方向にしか動けないからX方向に）1足すか引くかして消して書き……，といったものであるが，今回はそんなことはしなかった。なんとカズシゲの位置もボールの位置も，すべてテーブルとして用意してあるのだ。つまり状態によってその配列を読みにいって，その内容を引っ張り出し(そこには座標が入っているのだ)，その値の位置に表示するのだ。

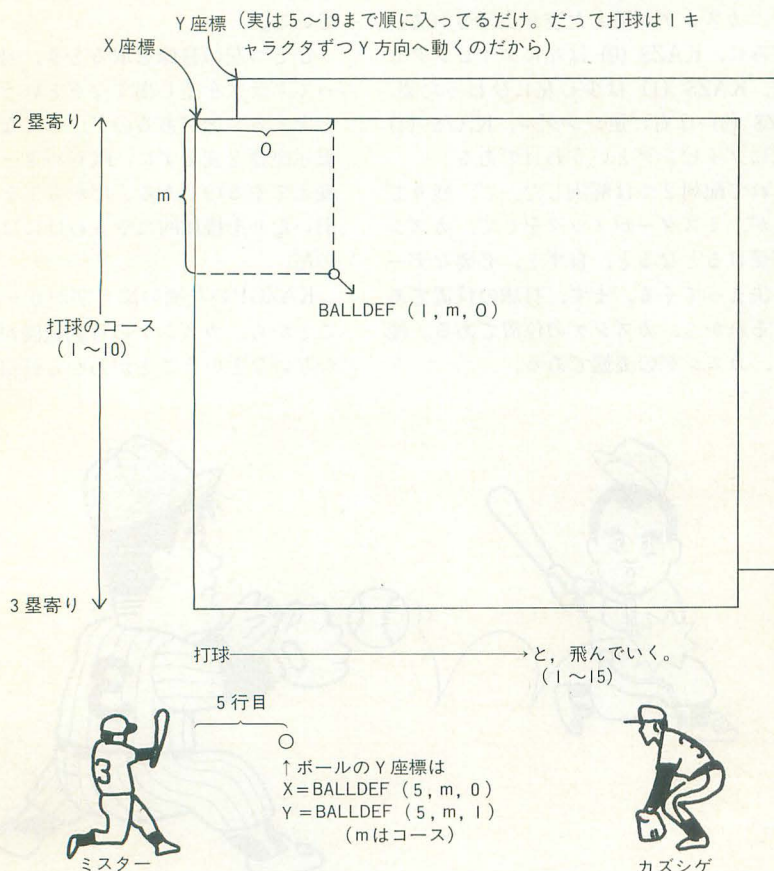
まず，KAZGPという2次元の配列がある。ここに入っているのは，カズシゲのグラフィック（PCGだけど）データである。PCGパターンには5つあるが，ある状態のとき，どのPCGを表示するかというデータが入っている。これは2番目の添え字が0のときである。2番目の添え字が1のほうには，カズシゲの表示されるX座標が入っている。

ひとつ左の打球を取るとき，身体をひねってグラブを差し出すなぞという不遜なことをすることがあるので，そんなときは，表示座標を変えずに，PCGパターンだけを変えてやるのである。だから1を足したり引いたりを機械的にやるわけにはいかないのだ。

KAZGPの左側の添え字が0～9であることから，カズシゲの移動範囲が10カ所しかないなということがわかる仕組みになっ

ている。カズシゲの現在位置がKAZGP(n,0)でわかるから，プレイヤーが“4”を押したら，ひとつ減らした配列のデータを，“6”だったらひとつ増やした配列のデータを読みにいってやればよろしいのだ。

PCGパターンによってグラブの位置が違うので，キャッチング判定のことも考える必要があったが，それは，表示X座標KAZGP (n,0) にPCGパターンNo.であるKAZGP (n,1) を足してやればよいように，キャラクタにスペースをくっつけたりして調整してある。このスペースは移動時に前のキャラクタを消す役割もあり，重宝している。

で，話は核心に少し近づいていくが，BALLPOSである。左側の添え字は一度に打球が4つまで（0を使って5つにしてもいいのだけれど，速度とか扱いやすさとかいろいろとあって，4にした）表示されることを表し，右の添え字は，0に打球の現在位置，1に打球の方向が入っている。これでミスターがノックした打球を管理しているのだ。

さて，READ文だけあって，肝心なDATA文がないBALLDEF (l,m,n) の登場である。ここが本プログラム最大のテーブルであって，3世代家族が一度に食事できるくらいである。

はじめに言ってしまうと，lは打球のY座標が入っている。ミスターとカズシゲの距離が15行とみなしての15である。2番目のmは打球の方向である。10パターンしかない。最後のnは，0のときX座標，1のときY座標を表している。ちなみに，lとmは0を使っていない。

というわけだったのだ。つまりこのミスターの打つ打球は，すべて，あらかじめ座標が決まっているのだった。

7 世界の終り
—2人の疲労—

ミスターのノックはいつも同じ高さで同じ速さのゴロばかりであり，しかも打球のコースさえ決まっていた。

親父も，マスコミに追い回されてすっかり現代人に染まってしまったな，と思ったカズシゲは悲しくなってきた。しかし，君も人のことは笑えない。そんな打球さえなかなか取れないのだから。微妙な動きのできないカズシゲにミスターは合わせているのかもしれない。

図 テーブルに置かれた打球のフルコース

しかし，父の体力的衰えもまた避けがたい事実であった。ノックのペースが速くなると，急に打球が遅くなるのだから。そのころには若いはずのカズシゲの動きも鈍くなっていたけれど。

8 ベーシック・ワンダーランド
—データの生成—

リスト3を見てほしい。これこそが，フフフ，秘密の3次元配列BALLDEFを埋めるプログラムである。これを一度実行すれば，どどーっとDATA文が15行生成され，それが“KAZ.TBL”なるファイルになるのだ。そのファイルをリスト1にマージしてやれば，丸く収まる。

どうして始めからDATA文にしないかって？ それは，あなた，私はこうしてDATA文を作ったからであり，そしてなにより読者の皆さんも無味乾燥な数字の列よりも，プログラムを打ち込んだほうが面白いはずだ。計算式をリスト1に埋め込むことも考えたけど，それでは毎回立ち上げるごとに同じ計算をすることになるから，いい気分がしない。

さて，リスト3である。

まず，打球のスタート位置。つまり，ミスターの位置と，打球が飛んできてほしい位置，つまりカズシゲのグラブの取り得る位置を配列に入れる。

続いて，グラブの取り得る位置10カ所それぞれについて，ミスターの位置とを結ぶ直線の式を作り，その式に従って，必要なY座標（打球は1行ずつ飛んでいく）から，それに対応するX座標を計算しているのである。

2つの座標を仮に，(x1,y1) と (x2,y2) とすると，この2点を結ぶ直線の式は，

y＝((y2−y1)/(x2−x1)) x＋(x1*(y2−y1)/(x2−x1))

となる。簡単な数学である。野球一筋のカズシゲだってわかるはずだ。これから，yの値を増やしながらxの値を計算しては配列に放り込んでいるのだ。

それでもって，あとで30100行から始まるDATA文を作ってディスクに書き込んでいるのである。

こうすると，リスト1のほうでは座標計算も余計な気づかいもなく，ただ配列をアクセスしてその値に応じて表示していけばいいのである。この手はいろいろと応用がきくことだろう。

メモリはある限り湯水のように使え！だ。

もっとも，配列も多次元になると少々アクセスに時間がかかるので，そのへんはうまく使ってほしい。あんまり気になるほどではないけどね。

9　世界の終り
―千本ノック?―

ミスターは千本ノックだといって，本気で（彼は単純だったから）1000本ノックする気でいた。

両者へとへとになりながら，ミスターは「千本なんていうんじゃなかった」と思い，カズシゲは「うちの監督が好々爺関根でよかった」と思った。

ただ数多く努力すればいいというものでもないのである。世の中のすべてにおいて，正しいわずかな努力は，間違ったたくさんの努力より多くの成果が見込めるのだ。要はどれだけ努力するかではなく，どう努力するかなのだよ。受験生諸君。

10　ベーシック・ワンダーランド
―守備率の表示―

最初，千本ノックと称して1000回ノックを受けるまで終わらないようにしようかと思ったが，あまりにも無謀であることがテストプレイ中に発覚し，100本に減らした（テストプレイは重要ですぞ）。 100本のノックを受けたあと，そのキャッチング率によりミスターの講評が聞けるので，努力精進を重ねてほしい。

＊　　＊

（注：当然ですが，この話はフィクションであり，登場人物その他が実在の人物と似ていたにしても，それは偶然である。実在の人物とはきっとなんの関係もない）

・あとがき

毎年3月も終わりになると，プロ野球選手名鑑が各社から発売されるのが楽しみで，シーズンが終わったあとからわくわくしている。私は日刊スポーツの文庫サイズのヤツを10年来愛用している。これは安くてよい。

なにを隠そう学生のころは，開幕前の選手名鑑をと思った途端，その前に控える試験やらレポートやら，ときには受験を思い出し，いやーな気分で落ち込み，早く来い来シーズンと，永遠に来るな試験期間の板挟みで苦しんだものだ。あの複雑な気分は今も忘れない。卒業した今，晴れて選手名鑑発売を待つことができる。やったね。

うーん，いわゆる
ワン サウザンド本
ノック　ですか

守備率は　32 %ですかぁ

これでは高校生以下ですね

TRY AGAIN ? (Y/N)?

参考文献

村上春樹,「世界の終りとハードボイルド・ワンダーランド」，新潮社

「88年度版　プロ野球選手名鑑」，日刊スポーツ

ゲームおよびプログラムの簡単な説明

まず，リスト2を打ち込んで実行し，リスト3を打ち込んで実行し，リスト1を打ち込んでリスト3で生成した“KAZ.TBL”をマージして実行すれば，遊べます。

ミスターのひと言がでかく表示され，待つこと3秒（待たないようにPAUSE文を取ってもいいけど）。「いくぞ，カズシゲ」の声（右上に表示される）とともに，ミスターのノックが始まります。

プレイヤーは左右に“4”と“6”でカズシゲを操りながら，ダイヤモンドグラブ目指してノックを100本受けます。終われば，ミスターのひと言が待っています。気をつけるべきは，カズシゲの位置とグラブの関係です。グラブの位置で打球を待ち受けないと，キャッチしたことにはなりません。正面を向いているときには身体の中心でかまわないのですが，左を向いているときには左で，右を向いているときには右で打球を受けてください。実は，身体で受けたときにはカズシゲが「うっ」とうめいたり，ミスターが「よく身体で止めた」などと褒めたりしたかったのですが，やめました。暇な方はいろいろと改造してください。

GOTO文はひとつしかないし，行番号にもまったく依存しないプログラムになっているのが，今の時代を表していますね。ただ私がBASIC以外の言語ばかり触っていたという事情もありますが。

結構，難しいので，ミスターの奴，卑怯だ！　と，思うこともあるでしょうが，怒らないでやってください。露骨にピコピコゲームなのですから。

リスト1　がんばれ！　カズシゲ君（BASIC-M25）

```
1000 '
1010 ' 頑張れ 一茂くん！！！！
1020 '                                  by K.Ogikubo
1030 '
1040 INIT "CRT2:320,200,16"
1050 INIT "CRT:40,25,0,0"
1060 WIDTH 40,25:KLIST 0
1070 CLEAR
1080 CLS 3
1090 DIM BALLDEF(15,10,1)
1100 DIM BALLPOS(4,1)
1110 DIM KAZ_GP(9,1)
1120 DIM MSG$(2)
1130 DIM SIGMSG$(3,2)
1140 DIM KAZ$(4)
1150 DIM SIG$(2)
1160 GOSUB *INIT
1170 *MAIN
1180 REPEAT ON ,4:CLICK OFF
1190 BALL_MAX=0:LEVEL=0:HIT_TM=13:CNTFLG=-1
1200
1210 WHILE CATCH+MISS<=100
```

```
1220    GOSUB *BALL_MOVE
1230    KEY 0,"":K$=INKEY$
1240    IF K$="4" THEN
1250       GOSUB *LEFT
1260    ELSE IF K$="6" THEN
1270       GOSUB *RIGHT
1280    END IF
1290    GOSUB *JUDGE
1300    KEY 0,"":K$=INKEY$
1310    IF K$="4" THEN
1320       GOSUB *LEFT
1330    ELSE IF K$="6" THEN
1340       GOSUB *RIGHT
1350    END IF
1360 WEND
1370 GOSUB *FIN
1380 IF CNTFLG THEN GOSUB *INIT2:GOTO *MAIN
1390 REPEAT ON ,2:CGEN 0:CLICK ON:WIDTH 80,25:END
1400 '
1410 *BALL_MOVE
1420 '
1430 IF (SIG_CHR=0 AND BALL_MAX<4) OR BALL_MAX=0 THEN
1440    HIT_TM=HIT_TM+1
1450    IF HIT_TM>14-LEVEL*2 THEN
1460       LINE (217,19)-(315,81),0,BF
1470       SYMBOL (218,20),"いくぞぉ!一茂",1,1,10,0,,1
1480       LOCATE 22,2:SIG_CHR=1:PRINT SIG$(SIG_CHR):BEEP
1490       HIT_TM=0:BALL_MAX=BALL_MAX+1:BALLPOS(BALL_MAX,1)=INT(RND(1)*10+1)
1500       BALLPOS(BALL_MAX,0)=1
1510       IF BALL_MAX<=1 THEN RETURN
1520    END IF
1530 ELSE IF SIG_CHR=1 THEN
1540    LOCATE 22,2:SIG_CHR=2:PRINT SIG$(SIG_CHR)
1550 ELSE IF SIG_CHR=2 THEN
1560    LOCATE 22,2:SIG_CHR=0:PRINT SIG$(SIG_CHR)
1570 END IF
1580 '
1590 *BALL_MOV2
1600 '
1610    FOR I=1 TO BALL_MAX
1620       BX=BALLPOS(I,0):BY=BALLPOS(I,1)
1630       LOCATE BALLDEF(BX,BY,0),BALLDEF(BX,BY,1):PRINT " ";
1640       BALLPOS(I,0)=BALLPOS(I,0)+1:BX=BALLPOS(I,0)
1650       LOCATE BALLDEF(BX,BY,0),BALLDEF(BX,BY,1):PRINT BALL$;
1660       IF BX=15 THEN MISS=MISS+1:GOSUB *ERRRTN:ERFLG=-1
1670    NEXT
1680    IF ERFLG THEN GOSUB *ZIGO
1690 RETURN
1700 '
1710 *ZIGO
1720 '
1730    FOR L=1 TO BALL_MAX-1
1740       BALLPOS(L,0)=BALLPOS(L+1,0):BALLPOS(L,1)=BALLPOS(L+1,1)
1750    NEXT:BALL_MAX=BALL_MAX-1:ERFLG=0
1760 RETURN
1770 '
1780 *JUDGE
1790 '
1800 FOR I=1 TO BALL_MAX
1810    IF BALLPOS(I,0)=14 AND BALLDEF(14,BALLPOS(I,1),0)=KAZGP(KAZ_X,0)+KAZGP(
KAZ_X,1) THEN
1820          MSG$(0)="グッドですよ":MSG$(1)="一茂君!":MSG$(2)=""
1830          GOSUB *MSGWRT
1840          LOCATE KAZGP(KAZ_X,1),REAL_Y:PRINT KAZ$(KAZGP(KAZ_X,0))
1850          LEVEL=LEVEL+.1:CATCH=CATCH+1:PLAY "@705L16A<A>A<A":PAUSE 1
1860          GOSUB *ZIGO
1870    END IF
1880 NEXT
1890 CGEN 0:LOCATE 25,20:PRINT "CATCH = ";CATCH:CGEN 1
1900 RETURN
1910 '
1920 *LEFT
1930 '
1940 IF KAZ_X=0 THEN RETURN
1950    KAZ_X=KAZ_X-1
1960    LOCATE KAZGP(KAZ_X,1),REAL_Y:PRINT KAZ$(KAZGP(KAZ_X,0))
1970 RETURN
1980 '
1990 *RIGHT
2000 '
2010 IF KAZ_X=9 THEN RETURN
2020    KAZ_X=KAZ_X+1
2030    LOCATE KAZGP(KAZ_X,1),REAL_Y:PRINT KAZ$(KAZGP(KAZ_X,0))
2040 RETURN
2050 '
2060 *MSGWRT
2070 '
2080 LINE (215,19)-(318,85),0,BF
2090 FOR MM=0 TO 2
2100    SYMBOL (218,MM*21+20),MSG$(MM),,,,,,1
2110 NEXT
2120 RETURN
2130 '
2140 *ERRRTN
2150
```

▶「OS-9/X68000」が29,800円。目からウロコが落ちた。なに「C&プロフェッショナルパッケージ」が58,000円？ うーむ，見事な攻撃。 安藤 康則 (24) 大阪府

```
2160 CASE=INT(RND(1)*4)
2170 MSG$(0)=SIGMSG$(CASE,0):MSG$(1)=SIGMSG$(CASE,1):MSG$(2)=SIGMSG$(CASE,2)
2180 GOSUB *MSGWRT:PAUSE 1
2190 LOCATE KAZGP(KAZ_X,1),REAL_Y:PRINT KAZ$(KAZGP(KAZ_X,0))
2200 BALLPOS(1,0)=0
2210 CGEN 0:LOCATE 25,21:PRINT "MISS = ";MISS:CGEN 1
2220 RETURN
2230 '
2240 *FIN
2250 '
2260 CLS 3:CGEN 0
2270 IF CATCH<30 THEN
2280    KOUHYOU$="ほんとに， 私の息子ですか？"
2290 ELSE IF CATCH<40 THEN
2300    KOUHYOU$="これでは高校生以下ですね"
2310 ELSE IF CATCH<50 THEN
2320    KOUHYOU$="6大学レベル，ですか？"
2330 ELSE IF CATCH<60 THEN
2340    KOUHYOU$="宇野か 原にも 負けてしまいますね
2350 ELSE IF CATCH<70 THEN
2360    KOUHYOU$="プロならば， 当然 ですね"
2370 ELSE IF CATCH<80 THEN
2380    KOUHYOU$="ダイヤモンドグラブ ，ですか？
2390 ELSE IF CATCH<90 THEN
2400    KOUHYOU$="いわゆる  名手，ぐっどぷれいやー
2410 ELSE
2420    KOUHYOU$="栄光の背番号3，ですか.思い出しますね
2430 END IF
2440 SYMBOL (0,50),"守備率は "+STR$(CATCH)+" %ですかぁ ,1,1,14,0,,1
2450 SYMBOL (0,90),KOUHYOU$,1,1,14,0,,1
2460 REPEAT ON ,2
2470 LOCATE 5,23:INPUT "TRY AGAIN ? (Y/N) ";TA$
2480 IF TA$="Y" OR TA$="y" THEN CNTFLG=-1 ELSE CNTFLG=0
2490 REPEAT ON ,4:RETURN
2500 '
2510 *INIT
2520 '
2530 FOR I=0 TO 1  :FOR J=0 TO 9:READ KAZGP(J,I):NEXT:NEXT
2540 FOR I=1 TO 15
2550    FOR J=1 TO 10
2560       READ BALLDEF(I,J,0),BALLDEF(I,J,1)
2570    NEXT
2580 NEXT
2590 KAZ$(2)=CHR$(32,48,49,50,32,31,29,29,29,29,29,32,51,52,53,32)
2600 KAZ$(1)=CHR$(32,54,55,56,32,31,29,29,29,29,29,32,57,58,59,32)
2610 KAZ$(3)=CHR$(32,60,61,62,32,31,29,29,29,29,29,32,63,64,65,32)
2620 KAZ$(0)=CHR$(66,67,68,69,31,29,29,29,29,70,71,72,73)
2630 KAZ$(4)=CHR$(32,74,75,76,77,31,29,29,29,29,29,32,78,79,80,81)
2640 SIG$(0)=CHR$(34,35,31,29,29,36,37)
2650 SIG$(1)=CHR$(38,39,31,29,29,40,41)`
2660 SIG$(2)=CHR$(42,43,31,29,29,44,45)
2670 BALL$=CHR$(47)
2680 *INIT2
2690 PAINT (0,0),4
2700 SYMBOL (10,20),"うーん，いわゆる",2,2,15,0,,1
2710 SYMBOL (40,70),"ワン サウザンド本 ",2,2,15,0,,1
2720 SYMBOL (60,120),"ノック  ですか ",2,2,15,0,,1
2730 PAUSE 3:CLS 3
2740 SIGMSG$(0,0)="ウェストが ":SIGMSG$(0,1)=" 高いですね ":SIGMSG$(0,2)="
2750 SIGMSG$(1,0)="もっとですね ":SIGMSG$(1,1)="打球に執着  ":SIGMSG$(1,2)="です
ね"
2760 SIGMSG$(2,0)="いわゆる ":SIGMSG$(2,1)="ヘッドアップ ":SIGMSG$(2,2)="ですか
2770 SIGMSG$(3,0)="野性の勘・・と ":SIGMSG$(3,1)="いいますか  ":SIGMSG$(3,2)="欲し
いですねぇ"
2780 PAINT (0,0),4
2790 LINE (190,16)-(190,199),15
2800 LINE (0,16)-(190,16),15
2810 LINE (182,192)-(190,200),15,BF
2820 LINE (6,16)-(14,24),15,BF
2830 LINE (182,20)-(186,24),0,BF
2840 '
2850 SIG_CHR=0:KAZ_X=4:REAL_Y=18:MISS=0:CATCH=0
2860 '
2870 LOCATE 25,20:PRINT "CATCH = ";CATCH
2880 CGEN 1
2890 LOCATE 22,2:PRINT SIG$(0)
2900 LOCATE KAZGP(4,1),18:PRINT KAZ$(KAZGP(4,0))
2910 '
2920 RETURN
2930 'KAZ DATA
2940 DATA 0,1,1,2,2,2,2,3,3,4
2950 DATA 15,15,16,16,17,18,19,19,20,20
```

リスト2　PCGデータ

```
20000 REM PCG EDITER DATA FILE
20010 DATA 0020,000000000000000000000000000000000000000000000000
20020 DATA 0021,000000000000000000000000000000000000000000000000
20030 DATA 0022,0000000102040C0E00010306060000020001030606000002
20040 DATA 0023,000020F030101030C0E0E07030101030C0E0E07030101030
20050 DATA 0024,060703000000000000010100000000000001010000000000
20060 DATA 0025,30F0E0000000000030F0E0000000000030F0E00000000000
```

▶俺は13歳にして，X68000を持っているぞー。　　井川　直己（13）岐阜県

```
20070 DATA 0026,00000003060C0C0E00000003060C0C0E00000003060C0C0E
20080 DATA 0027,000000F038181C3C000000F038181C3C000000F038181C3C
20090 DATA 0028,0603010160600000080F00F60600000080F00F60600000
20100 DATA 0029,38F8D880000000002008989800000000200898800000000
20110 DATA 002A,00000003060 40C0E0000010306040C0E00000003060 40C0E
20120 DATA 002B,000070F8381C18380000F0FB3B1F1226000070FB3B1F1226
20130 DATA 002C,06030300000000000600000000000000060000000000000
20140 DATA 002D,38F0FC780000000024440C0818000000024440C080000000
20150 DATA 002E,002070200000000000207020000000000002070200000000
20160 DATA 002F,00606000000000000006060000000000000606000000000
20170 DATA 0030,00000207060 60F0D030705000000010303070500000000101
20180 DATA 0031,003C3C7E7E7EFFFF808080000000858580B4A04200008585
20190 DATA 0032,00303038387070B0003000000000080C00030000000000080
20200 DATA 0033,0303031D191D0E0707031B1D1D1F0D07030303 1D191D0E07
20210 DATA 0034,FF81FEC1FE81AAFFC381FEC1FE81FFFFC381FEC1FE81AAFF
20220 DATA 0035,E0C0803030F06080E0C0D8B8B8F8F0E0E0C0803030F06080
20230 DATA 0036,00000010381C0E0DC8F8F860000000103C8F8F8600000000 1
20240 DATA 0037,00787CFEFE7F7FFF00000000000501C3007840C0800501C3
20250 DATA 0038,0080C0E06080E0C00080000000E0E0C000800000000 80E0C0
20260 DATA 0039,030301030F0F0F07070301050D0F0E070303010 30F0F0F07
20270 DATA 003A,F18EF0CEF18AAFFCF18EF0CEF18FFFFEF18EF0CEF18AAFFC
20280 DATA 003B,C0808080FC24FA0CC0808080FCFEFE06C0808080FC24FA0C
20290 DATA 003C,0000000000000000000000000000000010000000000000000
20300 DATA 003D,000E1E3F7F7FFFBC0000000000A08080000E020301208080
20310 DATA 003E,00000070E0C00008000F1F0F18000008000F1F0F18000008
20320 DATA 003F,01010101 3D153707010101617E7F7507010101013D153707
20330 DATA 0040,CEF1EEB1CE51F4FCCEF1EEB1CEF1FEFCCEF1EEB1CE51F4FC
20340 DATA 0041,38B8A080808080080C0C08080808000008080808080808000
20350 DATA 0042,00000000000001010000000000000008000000000000000180
20360 DATA 0043,F8F818041FFEFDFD8080001C1F06010180800004 1FC68101
20370 DATA 0044,00000000E8BF4F5F00000000F8BF5F5F00000000E8BF4F5F
20380 DATA 0045,000000000FCF8000000000000F3F70000000000000F0F000
20390 DATA 0046,01181F0700000000E0E060000000000 0E0E060000000000
20400 DATA 0047,FDBDDFEF00000000001453F1F000000000 01051F0F00000000
20410 DATA 0048,48DFEFFD00000000058DFFFFF00000000048DFEFFD00000000
20420 DATA 0049,00F8FC000000000000F7F38000000000000F0F0000000000 0
20430 DATA 004A,0000000003F1F0000000000000CFEF000000000000000F0F00
20440 DATA 004B,0000000017FFF3FA000000001FFFFBFA0000000017FFF3FA
20450 DATA 004C,00000037F7FFBFBF00000038F8E080800000000030F0E38180
20460 DATA 004D,000000C0F838808000000000307070000000000000307078000
20470 DATA 004E,001F3F0000000000000EFCF010000000000000F0F000000000 0
20480 DATA 004F,12FAF5BF000000001AFAFDFF0000000012FAF5BF00000000
20490 DATA 0050,BFBE7CF00F3F000080A07CF81101000080A07CF001010000
20500 DATA 0051,80000000000000000000000000000000000000000000000 0
20510 FOR I=1 TO 50 :READ C$,D$:DEF CHR$(VAL("&H"+C$))=HEXCHR$(D$):NEXT
```

リスト3 テーブル・ジェネレータ

```
1000 ' TABLE GENERATER                     for KAZUSHIGE
1010 '                                     by K.Ogikubo
1020 DIM POSDT(15,10,1)                :'z->0:X,1:Y y->左右 x->前後
1030 DIM DEFDT(10,1)                   :'x->POS y->0:X,1:Y
1040 F_NAME$="KAZ.TBL"
1050 IX1=1:IY1=1:IZ1=1
1060 FOR I=0 TO 10
1070     FOR J=0 TO 1
1080         READ A:DEFDT(I,J)=A
1090     NEXT
1100 NEXT
1110 OPEN "O",#1,F_NAME$
1120 DATA 22,2,15,18,16,18,17,18,18,18,19,18,20,18,21,18,22,18,23,17,24,17
1130 '   計算
1140 FOR I=1 TO 10
1150      IF DEFDT(0,0)<>DEFDT(I,0) THEN
1160          KATAMUKI=(DEFDT(I,1)-DEFDT(0,1))/(DEFDT(I,0)-DEFDT(0,0))
1170          KOUTEN=DEFDT(0,1)-KATAMUKI*DEFDT(0,0)
1180          FOR J=5 TO 19
1190              X=(J-KOUTEN)/KATAMUKI
1200              POSDT(J-4,I,0)=INT(X):POSDT(J-4,I,1)=INT(J)
1210          NEXT
1220      ELSE
1230          FOR J=5 TO 19
1240              X=DEFDT(0,0):POSDT(J-4,I,0)=INT(X):POSDT(J-4,I,1)=INT(J)
1250          NEXT
1260      END IF
1270 NEXT
1280 PRINT "計算終了．これから書き込みます．
1290 FOR I=1 TO 15
1300     GN=30100+I*10:DTS$=STR$(GN)+" DATA "
1310     FOR J=1 TO 9
1320         FOR K=0 TO 1
1330             DTS$=DTS$+STR$(POSDT(I,J,K))+",
1340         NEXT
1350     NEXT :DTS$=DTS$+STR$(POSDT(I,J,0))+","+STR$(POSDT(I,J,1))
1360     PRINT #1,DTS$:PRINT DTS$
1370 NEXT
1380 CLOSE #1
1390 END
```

会話プログラムへの道
まずは単語を見分けよう

Mounai Toshiyuki
毛内 俊行

コンピュータとおしゃべりする。その要は言語解析です。まず単語の役目とつながり方の規則について考えてみましょう。いずれはHALだってあなたのものですよ。

誰でも一度はパソコンで会話プログラムを作ろうと思ったことがあるでしょう。コンピュータと話をやり取りする，そんな興奮をどこかで経験した人もいると思います。しかし，会話プログラム作成への道は長く険しく，そしてとても迷いやすいのです。そこで，そうした迷い人のために，ひとつの道標を置いてみたいと思います。

BASICは，いわばパソコンユーザーにとって必修科目の言語です。それだけに誰でも気軽に触れる言語でもあります。そのBASICを使って，日本語の文章の解析を，ごく初歩的な段階で試みてみましょう。

プログラムはBASIC-M25(MZ-2500)で作りましたが一般的な命令しか使っていません。X1/turboならほとんど変更なしに動きますし，他の機種でも簡単に移植できるはずです。ただしMZ-2500では，日本語入力モードにして全角文字での入力が可能で，この場合，文字データを一度半角文字に変換してから処理します。他機種の場合はリスト1の200行を削除してください。

リスト1：共通プログラム

文章の解析は，会話プログラムを作る上で最初に突き当たる，そして最も大きな壁です。この作業をいいかげんにすると，プログラム作成は迷宮入りになりかねません。

これまでにも，雑誌などでいろいろな会話プログラムが発表されてきました。たいていの場合アルゴリズムは同じで，入力された文章からキーワードを作成し，返す文章を検索して返答する，というものです。

```
run
>ﾜﾀｼﾊ ｶﾚｦ ﾐﾀ
TALK$  =ﾜﾀｼﾊ ｶﾚｦ ﾐﾀ
WD$(1) =ﾜﾀｼﾊ
WD$(2) =ｶﾚｦ
WD$(3) =ﾐﾀ

>
```

リスト1＋リスト2実行例

キーワードが文意を解く鍵になるわけですから，入力された文章の解析がいかに重要かわかると思います。これから行うのはその鍵をピックアップすることです。

では，キーワード探索名所巡りの旅に出発しましょう。準備として，文章の入力などを行う手順が必要です。リスト1を入力してください。これは今後入力するプログラムすべての共通部で，文章の入力を受け，文を構成する各語に分解して，それらをWD$という配列変数に代入するものです。

この共通プログラムは，他の6つのプログラムとMERGEして使います。

リスト1　共通プログラム

```
100 '--------------------------------
110 '          入力 (LIST-1).
120 '--------------------------------
130 DEF INT A-Z
140 OPTION BASE 1
150 WMAX=20                        ' WMAX=文章中の最大単語数
160 DIM WD$(WMAX),ANS$(5)
170 '
180 *INP
190 INPUT ">",TALK$                ' 文章の入力
200 TALK$=KACNV$(TALK$)            ' 全角→半角
210 IF TALK$="" THEN 190
220 '
230 TK$=TALK$
240 FOR I=1 TO WMAX
250   A=INSTR(TK$," ")
260   IF A=0 THEN WD$(I)=TK$:GOTO 310
270   WD$(I)=LEFT$(TK$,A-1)
280   TK$=MID$(TK$,A+1)
290 NEXT I
300 I=I-1
310 WE=I                           WE=文章中の単語数
320 PRINT "TALK$  =";TALK$
330 '
```

リスト2　基本動作を確かめる

```
500 '--------------------------------
510 '          解析 (LIST-2)
520 '--------------------------------
530 *KAISEKI
540 FOR I=1 TO WE
550   PRINT "WD$(";MID$(STR$(I),2);") =";WD$(I)
560 NEXT I
570 PRINT
580 GOTO *INP
```

リスト2：基本動作

次にリスト1の働きを見てみましょう。リスト2を追加して実行してみてください。すぐに>が出て文章の入力待ちになりますから，試しに，

　ワタシハ　カレヲ　ミタ

と入力してみてください。3つの語がそれぞれ配列変数WD$に代入されて画面に表示されます。

リスト1＋リスト2
(入力された文を各語に分け，配列変数に入れる。語数は最大20字まで)

　ワタシハ　カレヲ　ミタ
　変数1 ＝ ワタシハ
　変数2 ＝ カレヲ
　変数3 ＝ ミタ

なお，よく見られる会話プログラムでは，

　ワタシ　ハ　カレ　ヲ　ミタ

と，助詞などを切り離して入力するものが多いのですが，これは個人的にあまり好きではないので，ここでは切り離さないでください。

リスト3：長い語はあやしい

文章からその文意を示すキーワードを選び出すには，それなりの手掛かりが必要です。リスト3のプログラムは昔からよく使われてきた技法で,「文章中のなるべく長い語をキーワードにする」というものです。この方法は理論が単純でよく使われた技法なのですが，信頼性がとても低いという重大な欠点を持っています。リスト3では，長い語を2語まで拾って配列変数ANS$に代入します。このやり方だと，さきほどと同様に入力した場合，拾われる語は「ワタシハ」と「カレヲ」であり，述語の「ミタ」は無視されます。

リスト3
(長い語を2つ拾う)

　ワタシハ　カレヲ　ミタ
　変数1 ＝ ワタシハ
　変数2 ＝ カレヲ

これでは文意を把握するための重要なキーが抜けていることになりますね。おまけに

リスト3実行例

文章中の語数が多くなるほど，拾われたキーワードの示す意味がわからなくなる恐れもあります。

つまり，長い語は「あやしい」だけであって，それがキーワードになり得る根拠はどこにもありません。

そこで，述語に重点を置いてキーワード探しに工夫をしたのが，次に示すプログラムです。

リスト4：最後の単語もくさい

文章中の述語をキーワードのひとつとして拾ってくるプログラム，それがリスト4です。言語には，ある程度自由な構成が可能なところがあります。たとえば,「昨日私は駅で彼に会った」という文章は，少々単

リスト4実行例

語を入れ替えて「私は昨日駅で彼に会った」としても「私は彼に昨日駅で会った」としても意味は変化せず，また文章として正しいものです。そしてどの場合でも，述語である「会った」は，必ず文章の最後に位置しています。つまり，(日本語の)述語というものは文章の最後にくるものなのです(もちろん例外はある)。

これでルールが決まりました。2つのキーワードのうち，ひとつは文章の最後から拾ってくればいいのです。リスト4では，もうひとつのキーワードはリスト3と同様に長い語を拾ってきていますが，結果は格段によくなったはずです。

リスト4
(ひとつは最後にくる語を，もうひとつは長い語を拾う)

　ワタシハ　カレヲ　ミタ
　変数1 ＝ ワタシハ
　変数2 ＝ ミタ

リスト5：構文を考える

さて，考え方をもう少し進めてみましょう。「いつ(時間)，どこで(場所)，なにが(主語)，なにを(目的語)，どうした(述語)」という構文をもとに，キーワードを拾う方法を考えます。

リスト5では，上に挙げた構文の5つの動作のうち「いつ」と「どこで」については解析を行っていません。これらの動作については，リスト6のプログラムで扱います。

さて，主語・述語・目的語の判別に移りましょう。述語は前述したように最後の語から拾います。問題は主語と目的語の判定です。これは各語の語尾を見て行います。たとえば，文章中に「私は」という語があったとき，この語は「私」＋「は」に分解されます。助詞「は」は多くの場合主語に付属するので,「私は」は主語であると判断できるでしょう。一方,「は」の代わりに「を」を使い「私を」とすれば，これは目的語になります。

リスト3　長い語を拾う

```
500 '--------------------------------
510 '           解析 (LIST-3)
520 '--------------------------------
530 *KAISEKI
540 P1=1:L1=LEN(WD$(1)):P2=0:L2=0:ANS$(2)=""
550 FOR I=2 TO WE
560   LP=LEN(WD$(I))
570   IF LP<L1 THEN 590
580   P2=P1:L2=L1:P1=I:L1=LEN(WD$(I)):GOTO 600
590   IF LP=>L2 THEN P2=I:L2=LEN(WD$(I))
600 NEXT I
610 IF P1>P2 AND P2<>0 THEN SWAP P1,P2
620 ANS$(1)=WD$(P1)
630 IF WE<>1 THEN ANS$(2)=WD$(P2)
640 PRINT "ANS$(1)=";ANS$(1)
650 PRINT "ANS$(2)=";ANS$(2)
660 PRINT
670 GOTO *INP
```

リスト4　述語は文末から

```
500 '--------------------------------
510 '           解析 (LIST-4)
520 '--------------------------------
530 *KAISEKI
540 ANS$(2)=WD$(WE):ANS$(1)=""
550 IF WE=1 THEN 610
560 P1=1:L1=LEN(WD$(1))
570 FOR I=2 TO WE-1
580   IF LEN(WD$(I))=>LEN(WD$(P1)) THEN P1=I
590 NEXT I
600 ANS$(1)=WD$(P1)
610 PRINT "ANS$(1)=";ANS$(1)
620 PRINT "ANS$(2)=";ANS$(2)
630 PRINT
640 GOTO *INP
```

リスト5

（語尾から主語と目的語を判断する。述語は文末から拾う）

　ボクハ オトトイ カノジョヲ エキデ ミタ
　何が　　：ボクハ
　何を　　：カノジョヲ
　どうした：ミタ

こうして主語と目的語も判別できます。この方法は次に紹介する辞書を使った解析法でも有効で，さらにリスト7では，リスト6だけでは扱いきれなかった単語について，この方法を用いて判断しています。

リスト6：最終兵器の単語辞書

単語辞書をプログラム中や変数内にあらかじめセットしておき，入力された文章の構成語をその辞書で解析する，というプログラムがリスト6です。

文章中の単語は，すべてこの辞書によりまず名詞か動詞かの判断をして，それから主語・述語・目的語のふり分けを行います。ふり分ける内容は，一般名詞が主語あるいは目的語，動詞が述語となり，目的語の区別は助詞などの語尾により行います。ですから下の例に挙げたように，述語が文の最後にきていないような場合でも，それが辞書にあれば述語として判断されます。

リスト6

（辞書により各語を判定する）

　ボクハ オトトイ エキデ ミタ カノジョヲ
　いつ　　：オトトイ
　どこで　：エキデ
　何が　　：ボクハ
　何を　　：カノジョヲ
　どうした：ミタ

この辞書では，類似の性質を持つものをまとめました。たとえば，一人称代名詞（プログラムでは一般名詞としている）の「ワタシ，ボク，ワタシタチ」をひとつのグループにし，また動詞の場合は活用形をひとまとめにしています。プログラムを実行したときに，表示される各キーワードの後ろの（　）内に出るのは，辞書の中で同じグループに入っている単語の例です。

こうしたデータを拡充すれば，さらに広範で正確な判別ができるようになるでしょう。皆さんでどんどん追加してみてください。

しかし，この方法では辞書の中にある単語だけで会話をしなくてはならない，という制限があります。そこで，辞書にない単語についても少しは考えてみようとしたのがリスト7です。

リスト7：より賢く

これはリスト6にさらに追加して用います。語尾を調べて，少なくとも目的語と述語を見つけ，キーワードとして判定するものです。辞書にない単語だらけの文章でも，目的語と述語がわかればなんとか大意はとることができます。

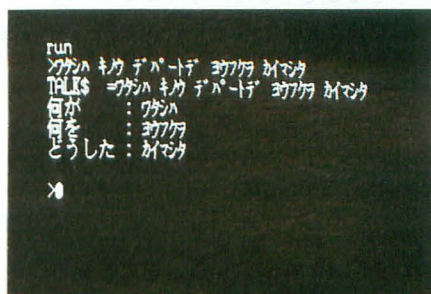
リスト6＋7

（辞書にない語も語尾で判定する）

　カレハ キノウ ココデ リンゴヲ タベタ
　いつ　　：キノウ
　どこで　：ココデ
　何が　　：カレハ
　何を　　：リンゴヲ
　どうした：タベタ

さて，これで辞書にない単語があっても少しはなんとかできるようになりました。しかし，これで完璧というわけではもちろんありません。今回はあえてごく初歩的なものにとどめました。それは実際に文章を入力してみればわかると思います。

たとえば主語の判定。リスト6，7では辞書の持つ一般名詞以外の語が主語として入力された場合，それは主語として判定されません。では目的語と同様に，語尾にくる助詞によって主語も判定すればいいのではないでしょうか？　しかし，リスト5で使ったような助詞（ハ，ガ）は，必ずしも主語に付随するとは限らないのです。「私は，リンゴは好きだがバナナは嫌いだ」などのように，目的語に「は」がつく場合もあります。また「今日は天気がいい」なんてときはどうしましょう。こうした場合にキーワードを拾うには，さらに細かいルールが必要になるわけです。それは本格的な構文解析にもつながっていくでしょう。

私の道案内はここまで。この先は皆さんが自分で壁を突破していってください。自分で文章解析を試みれば，新たな技法も考えつくはずです。

リスト5　語尾で見分ける

```
500 '-------------------------------
510 '          解析 (LIST-5)
520 '-------------------------------
530 *KAISEKI
540 ANS$(1)="":ANS$(2)="":ANS$(3)=WD$(WE)
550 FOR I=1 TO WE-1
560   RESTORE *GOBI
570   REPEAT
580     READ CODE,GB$
590     IF GB$="*" THEN 620
600   UNTIL RIGHT$(WD$(I),LEN(GB$))=GB$
610   ANS$(CODE)=WD$(I)
620 NEXT I
630 PRINT "何が      : ";ANS$(1)
640 PRINT "何を      : ";ANS$(2)
650 PRINT "どうした : ";ANS$(3)
660 PRINT
670 GOTO *INP
680 '
1000 *GOBI
1010 DATA 2,ﾙﾄ,2,ｳﾄ,2,ﾃﾊ,2,ｯﾃ,2,ｶﾗ,2,ﾃﾞ
1020 DATA 2,ｦ,2,ﾆ,2,ﾍ,2,ﾅ,2,ﾃ,2,ﾄ
1030 DATA 2,ﾉ,2,ﾓ,1,ｶﾞ,1,ﾊ,0,*
```

リスト5実行例

リスト６＋リスト７実行例

記念の第一歩

私が以上をもとに作ろうと目指している会話プログラムは，パソコンを相手にお喋りするのが目的のものです。現在盛んにもてはやされている人工知能とは，当然その主旨も次元も異なります。巷ではいくつもの会話プログラムがお喋りをしており，なかにはBASICだけで書かれたものも存在します（しかもそれが８ビットマシンで動いていたりするという）。

ここでは，会話プログラムの中心部である文章の解析という問題について，その基本を紹介しました。ほかにもいろいろな技法がありますが，それらについては，皆さんが挑戦するうち自然に見えてくるでしょう。

余談ですが，現在支配的な言語理論のひとつは，「人間の言語は創造的であり，人は一度も耳にしたことのない文章を無限に作り，解釈することができる」と強調しています。この理論で仮定されているのが文の深層構造です。本質的な意味を決定する深層構造は，実際の表現である表層構造の根底を成し，また一定の規則によって，文から深層構造へと生成する過程をツリーで表すことができます。

文章の解析に首をつっこむと，こうしたさまざまな仮説に出会うでしょう。そして最後には，日本語という矛盾だらけの言語を改めて好きになるでしょうし，また私のように，学校でもっと真面目に国語を学ぶべきだったと反省するかもしれません（本当だよ)。会話をコンピュータにシミュレートさせることもまた，考えることを山のように提供してくれますね。

今回は肯定文を軸として行いましたが，疑問文や否定文も基本的には同じです。あとはやる気と根気だけ。H.A.L.9000とまではいかなくとも，自然に会話できるようなプログラム目指してがんばりましょう。私が示したことはすべて，そのための第一歩にすぎません。この第一歩をどこに向けるかも皆さん次第なのです。

リスト6　辞書バージョン

```
500 '--------------------------------
510 '          解析 (LIST-6)
520 '--------------------------------
530 *KAISEKI
540 GOSUB *KDIC
550 PRINT "いつ      : ";ANS$(1),"(";ANS2$(1);")"
560 PRINT "どこで    : ";ANS$(2),"(";ANS2$(2);")"
570 PRINT "何が      : ";ANS$(3),"(";ANS2$(3);")"
580 PRINT "何を      : ";ANS$(5),"(";ANS2$(5);")"
590 PRINT "どうした  : ";ANS$(4),"(";ANS2$(4);")"
600 PRINT
610 GOTO *INP
620 '
630 *KDIC
640 FOR I=1 TO 5:ANS$(I)="":ANS2$(I)="":NEXT I
650 FOR AP=1 TO WE
660   RESTORE *DIC
670   READ CODE,DICW$:DIC2$=DICW$
680   IF CODE=0 THEN 740 ELSE 700
690   READ DICW$
700   IF DICW$="*" THEN 670
710   IF LEFT$(WD$(AP),LEN(DICW$))<>DICW$ THEN 690
720   IF CODE=>3 THEN GOSUB *MCHK       '目的語のチェック
730   ANS$(CODE)=WD$(AP):ANS2$(CODE)=DIC2$
740 NEXT AP
750 RETURN
760 *MCHK
770 RESTORE *GOBI
780 REPEAT
790   READ GB$
800   IF GB$="*" THEN RETURN
810 UNTIL RIGHT$(WD$(AP),LEN(GB$))=GB$
820 CODE=5:RETURN
830 '
1000 *GOBI
1010 DATA ﾙﾄ,ｳﾄ,ﾃﾊ,ｯﾃ,ｶﾗ,ﾃﾞ
1020 DATA ｦ,ﾆ,ﾍ,ﾅ,ﾃ,ﾄ,ﾉ,ﾓ,*
1030 '
1040 '--------------------------------
1050 '            辞書
1060 '    code 1=時間を表す名詞
1070 '         2=場所を表す名詞
1080 '         3=一般名詞
1090 '         4=動詞になる単語
1100 '--------------------------------
1110 '
1120 *DIC
1130 DATA 1,ｷｮｳ,*
1140 DATA 1,ｱｼﾀ,*
1150 DATA 1,ｷﾉｳ,*
1160 DATA 1,ｵﾄﾄｲ,*
1170 DATA 1,ｲﾏ,*
1180 '
1190 DATA 2,ｺｺ,*
1200 DATA 2,ｴｷ,*
1210 '
1220 DATA 3,ﾜﾀｼ,ﾎﾞｸ,ﾜﾀｼﾀﾁ,*
1230 DATA 3,ｱﾅﾀ,ｷﾐ,ｵﾏｴ,ｷﾐﾀﾁ,ｵﾏｴﾀﾁ,*
1240 DATA 3,ｶﾚ,ｶﾉｼﾞｮ,ｶﾚﾗ,ｱｲﾂ,ｱｲﾂﾗ,*
1250 '
1260 DATA 4,ｽﾙ,ｼﾅｲ,ｼﾏｽ,ｽﾚﾊﾞ,ｼﾛ,ｼﾖｳ,ｼﾀ,ｼﾃ,*
1270 DATA 4,ｸﾙ,ｺﾅｲ,ｷﾏｽ,ｸﾚﾊﾞ,ｺｲ,ｺﾖｳ,ｷﾀ,ｷﾃ,*
1280 DATA 4,ｲｳ,ｲﾜ,ｲｲ,ｲｴ,ｲｵ,ｲｯﾃ,ﾊﾅｻ,ﾊﾅｼ,ﾊﾅｽ,ﾊﾅｾ,ﾊﾅｿ,ｼｬﾍﾞ,*
1290 DATA 4,ｲｸ,ｲｶ,ｲｷ,ｲｹ,ｲｺ,ｲｯﾃ,ｲｯﾀ,*
1300 DATA 4,ｱｿﾌﾞ,ｱｿﾊﾞ,ｱｿﾋﾞ,ｱｿﾍﾞ,ｱｿﾝ,ｱｿﾎﾞ,*
1310 DATA 4,ｱｳ,ｱﾜﾅ,ｱｲﾏ,ｱｴﾊﾞ,ｱｵｳ,ｱｯﾀ,ｱｯﾃ,*
1320 DATA 4,ﾓﾗｳ,ﾓﾗﾜ,ﾓﾗｲ,ﾓﾗｴ,ﾓﾗｵ,ﾓﾗｯ,*
1330 DATA 4,ﾜﾗｳ,ﾜﾗﾜ,ﾜﾗｲ,ﾜﾗｴ,ﾜﾗｵ,ﾜﾗｯ,*
1340 DATA 4,ﾈﾙ,ﾈﾑﾗ,ﾈﾑﾘ,ﾈﾑﾙ,ﾈﾑﾚ,ﾈﾑﾛｳ,ﾈﾅｲ,ﾈﾏｽ,ﾈﾅｲ,ﾈﾖｳ,ﾈﾛ,ﾈﾃ,ﾈﾑｯ,*
1350 DATA 4,ﾐﾙ,ﾐﾅｲ,ﾐﾚﾊﾞ,ﾐﾖｳ,ﾐｶｹ,ﾐﾂｹ,ﾐﾏｾ,ﾐﾏｼ,ﾐﾀ,*
1360 DATA 4,ｵﾓｳ,ｵﾓﾜ,ｵﾓｲ,ｵﾓｴ,ｵﾓｵ,ｵﾓｯ,*
1370 DATA 0,*
```

リスト7　プラスアルファ

```
540 GOSUB *KDIC:GOSUB *JPIC:GOSUB *MPIC
840 '----------(LIST-7 追加)---------
850 *JPIC
860 IF ANS$(4)<>"" THEN RETURN
870 ANS$(4)=WD$(WE):ANS2$(4)="---":RETURN
880 *MPIC
890 IF ANS$(5)<>"" THEN RETURN
900 FOR I=1 TO WE-1
910   RESTORE *GOBI
920   READ GB$:IF GB$="*" THEN 980
930   IF RIGHT$(WD$(I),LEN(GB$))<>GB$ THEN 920
940   FOR J=1 TO 2
950     IF WD$(I)=ANS$(J) THEN 920
960   NEXT J
970   ANS$(5)=WD$(I):ANS2$(5)="---":RETURN
980 NEXT I:RETURN
990 '
```

特集 BASIC“おもちゃ箱”

☆元気なオタッキー作法講座(予告編?) 穴掘りいくぞっ，おーっ！

いきなり「オタッキー」とか言われたって，一般人にはなんのことやらわかりませんよね。でも，お読みになれば「こーゆー奴なんだ」とご理解いただけるでしょう。

Komura Satoshi
古村　聡

まずはタイトルについて

だーかーらあ，私はオタッキーぢゃないって言ってるでしょ，もう。ただね，こういうプログラミング講座をやるのが夢だったんだよねー。だから（予告編）なんだけど。いつかやるんだ，そういう連載。オタッキー養成講座だったりして。なまんだぶ，なまんだぶ。

それはともかく，まあ，なんです。皆さん，ゲームしてます？　なに，今日もリボルティーでバリバリ？　おお，そりゃいいですねえ。べつにスペハリでも，マッピーでもなんだっていいんだけど，みんな結構お金も時間もかかるでしょうゲームやるのに。

んでね，今日のこのコーナーのテーマは「ゲームを遊ぶ暇少しだけ削って自作ゲームを作ってOh！Xに送ってゲーム代をひねり出そう」という崇高なテーマのもとに繰り広げられるわけなんだなこれが。題して「とにかく手早く遊べるゲームを作っちゃう法」。一応，X1のNewBASICでゲームを作りますけど，それ以外のユーザーの皆さんも読んでやってください。かるーい読み物ですんで（どーせ，読み物しか書けませんよーだ。ぶつぶつ）。

私はこうしてゲームをイメージした

さーて，こうした不純な動機でもってBASICを立ち上げるわけですけど，パソコンの前に座ったからといって，ゲームのアイデアなんてポコポコ出てくるわけじゃない。なに!?　出てくる？　モウッ，出てこないのっ！　ね，出てこないでしょ。で，出てこなかったところでどうするか。そういうときは目に頼るわけです。うちの学校の心理学の教科書によれば，人間の5つの感覚，味覚，触角，嗅覚，視覚，聴覚のうち，視覚に置かれているウエイトは80%だそーで，したがって，形として目に訴えるようなキャラクターのエディットやらPCGの定義やなんかを先にやってしまうのが心理学的にいちばん正しい暇プロのあり方ということになります。うーん，なかなかアカデミック。

教訓1　オタッキーたるもの，ゲームを作ろうと思ったらデザインから始めるべし

まずはdefchrtool（BASICに付属のPCGを定義するツール）を動かしてキャラクターを作ります。さーて，ここで要注意。あまり気合を入れすぎるようなことをしてはいけません。あくまでも暇プロ（かなり本当の定義とは違うような気がするけど）であるから本業に差し支えちゃいけないんだよね。

ま，それはともかく，私の場合，自分のいちばん好きなものからてきとーにキャラクターを作ってしまいました。これが凄まじいてきとーさなんですよ。まずはですねえ，やっぱりYsみたいなRPGにしたいなーなんて考えて草原と木と河を作る。それから私は基本的にSFが好きだもんで，横スクロールのゲームもいいなあなんていってMy Shipを描いてしまったりするわけです。ついでに☆なんかも描いてっと。あーあ，脈絡なんてありゃしないんだから。

でもこの作業が一番楽しいかもしれないですね。なーんにも考えずにただひたすらに絵を描くだけなんだから。キャラクターさえ出来てしまえばこっちのもんですからね（と作っている本人はそう思い込んでいる）[1)]。うさぎちゃん（まじゃべんちゃーに出てきたバニーガールchanですね）だっていいし，光るどんぶりだってなんだっていいんです。ああ，華麗なる趣味の世界。

☆とMyShipだけでもゲームになる

さて，ほとんど世も末のようなキャラクターが完成したところでこれをゲームとしてまとめようとするわけですが，我ながら頭が痛い。Ysタイプの草原と河が出てきて☆が飛び回りプレイヤーは宇宙船で敵がうさぎちゃんとどんぶりじゃゲームはできようはずがない。なぜって，どんぶりはともかく，ぼくにはうさぎちゃんは殺せない。基本的に女の子をいじめるのは私の性にあわない……。ん？　ちゃう，ちゃう，そーゆー問題ではなくて，ゲームのジャンルとしてひとつにまとまらないんですよね。

そこで私は考えた。今日と明日で2本のゲームを作ろっと。うーん，あたまいい，平成の（で）はひと味違うぜ！　じゃあ，まあ，今日はウォーミングアップということで一番お手軽な☆とMy Shipでいこう。

教訓2　やたらと女の子を出そうとするのはオタッキーではなく単なるドすけべ

気を取り直してと。☆ですか。んじゃ，あれだ。昔よくあったあれでいこう。ではまず，材料を用意します。てきとーなアセンブラかマシン語のコード表を用意します。続いて，下ごしらえ。PCGで作ったキャラクターをスクロールさせるルーチンを作ります。そして☆の表示ルーチン，My Shipの表示と移動を作り，最後にMy Shipの当たり判定を作って出来上がり。

1）こーいうのって，ただの勘違いと思うかもしれないけど，この勘違いがプログラミングのためのいちばんのエネルギーとなったりするんですよ。

あっという間のスクロールゲーム

わずか28行でできたのが，このゲーム。わかるよねえ，だいたい。要するに横スクロールで☆が流れてそれに当たらないように自機を2・8キーで上下によけるゲームなんだよね。

なに，今月はマシン語を使ったら反則じゃないかって？　あまい，あまい。5行以内の反則は反則ではないと，かのアブドーラ・ザ・ブッチャーもいっているのさ（いってない，いってない！）[2]。

教訓3　格言は先に言った者の勝ち

一応，説明しておくとマシン語のルーチンは右から左へ☆を移動させるためにテキストVRAMのあたま（0，0）座標からスクロールさせたい領域の最後の部分までを(今の番地の内容)＝（今の番地＋1の内容）というふうに転送している。で，そのあとアトリビュート[3]を転送しただけ。だから，左側に消えたと思った☆はまた右側から出てくる。で，☆は消えないのに次から次から出てきて増えちゃうからそりゃもう大騒ぎさっ。

で，この手のゲームっていうのは，かなり昔っからあったわけなんだけど，ゲームを作るためのプロセスの主な要素っていうのはほとんど含んでいるように思うんだよね。たとえば背景。ずーっと，☆が飛んでいく。あの，星の海のなかを飛んでみたいっていう誰でも考える願いっていうのが，実はあのゲームそのものでしょ。以前にも言ったことがあるけど，ゲームっていうのはその人の願いをかなえるものだって。だから好きなキャラクターを作ってそれを背景にするのは，まさしくゲームの上で夢をかなえようとするのと同じ行為なんだよね。

うーん，話が電脳遊戯民になってしまったところで，次は余ったキャラの使い方でも考えるとしましょうか。

願いは大金持ち！

んで，翌日。といってもねー。残りのキャラっていうのがかなりの難物なんだよねー。これでゲームをねー。うーん。えーい，こーゆーときは遊んでしまえーっ！

というわけで，友達のところへ押しかけてしまうわけです。「ねー，なんかいいアイデアなーい？」

ただし，彼女のいる奴のところへは遊びに行かないこと。たいていの場合，自分の家に帰ったころには「全世界の彼女いる奴抹殺ゲーム」になってしまいますんで（えー，なりませんか？）。

教訓4　悪友は生かすべからず殺すべからず

それで友達からアイデアを奪い取ってくるわけですが，やっぱり持つべきものは友ですよー，うん。

まー，世の中というのは本当にいろんなことを考えている人がいるもんで，いろんな意見を言ってくれます。いいアイデアから最低なのまで[4]。

さて，ある友達はこんなことを言いました。「画面の左側にさ，宝を埋めた迷路作っておいて右側に地図描くのってどう？」

おお，そりゃいいや。しかしみんな，面白いことを思いつくもんです。なんか，小学生のころを思い出しますねー。ノートと鉛筆があれば2人で本拠地と大砲を書いて大砲から鉛筆をピンってはじいてゲームをするっていう，あのノリだもんねー。紙に♯と○×の駒を作って3目をやるとかやったでしょ？　ゲーム作りの原点はああいうところにあったんですねー。だいたい子供っていうのはそこに何かがあればそれでなんかしら遊びを作っちゃうんですよね。ゲームを作れる条件というのは案外，「永遠の少年・少女でいること」なんていうことかもしれません。

さて，じゃ，左側に迷路を作るとしますか。適当なブロックで塗りつぶしてと。ゲームをRUNさせてみればわかりますがブロックが真っ青でしょ。実をいうとこのブロックはRPGの河を作ろうと思って作ったブロックなんですよね。あはははは。

教訓5　あとは野となれ山となれ

で，迷路といっても，自分で遊ぶために作るんだからコンピュータに自動生成させるしかない。まっとうなものを作っていたら時間がかかりそうだしそれに「もしかしたら再帰呼び出し[5]を使うのはどうかなー」などと考えてしまい，迷路を作るのではなく「てきとーな乱数で穴を掘る」というふうに方針を変更しました。どんどんいい加減になっていくようですが，これが楽しいプログラミングというものです。気にしないように。

ブロックの中に宝を埋めて……，あっと，

リスト1　あっという間のスクロール

```
10 WIDTH 40:CLEAR &HC000 :SX=20:SY=12
20 DEFCHR$(42)=HEXCHR$("00000000000000000818FF7E3C7E66810818FF7E3C7E6681")
30 DEFCHR$(62)=HEXCHR$("C0E0FEFFFFF660C0000000050000000000C0E0F8A9FFF660C0")
40 GOSUB 190
50 LOCATE0,24:PRINT"Score=";: SC=0 :'ｽｺｱ ﾀﾞﾖ
60 HY=INT(RND(1)*100)
70 IF HY>21 THEN 90
80 LOCATE 39,HY::CGEN1:PRINT"*";:CGEN0
90 LOCATE10,24:PRINTSC;
100 LOCATE SX  ,SY:PRINT" ";
110 CALL &HC000:SC=SC+1
120 AT$=SCRN$(SX,SY,1): IF At$="*" THEN 180
130 I$=INKEY$(0)
140 SY=SY+(SY<16)*(I$="2")-(SY>1)*(I$="8")
150 AT$=SCRN$(SX,SY,1)
160 LOCATE SX,SY:CGEN1:PRINT">";:CGEN0:IF AT$="*" THEN 180
170 GOTO 60
180 STOP
190 'ﾏｼﾝｺﾞ ｶｷｺﾐ
200 SUM=0:FOR  ADR=&HC000 TO &HC027
210 READ DA$:POKE ADR,VAL("&H"+DA$):SUM=SUM+VAL("&H"+DA$)
220 NEXT:IF SUM=3652 THEN RETURN
230 PRINT"ﾏｼﾝｺﾞ ﾌﾞ ﾌﾞ ﾝﾆｴﾗｰｶﾞ ｱﾘﾏｽ":STOP
240 DATA 21,01,30,11,48,03,44,4D
250 DATA ED,78,0B,ED,79,CD,1D,C0
260 DATA 03,ED,78,0B,ED,79,23,1B
270 DATA 7A,B3,20,EA,C9,78,D6,10
280 DATA 47,C9,00,00,00,00,00,00
```

ブロックの中だとブロックを掘るという動作がいるな。これは簡単，自分が歩くとブロックが消えるようにすればいいんだね，と。そうすれば，自分の回りが地図と違ってくるからプレイヤーは混乱するに違いない。意地悪っていうのはやりすぎると嫌われるけどほどほどにやる分にはゲームが面白くなると思う。て，いうより相手が全然意地が悪くなかったらゲームにならないですよね。

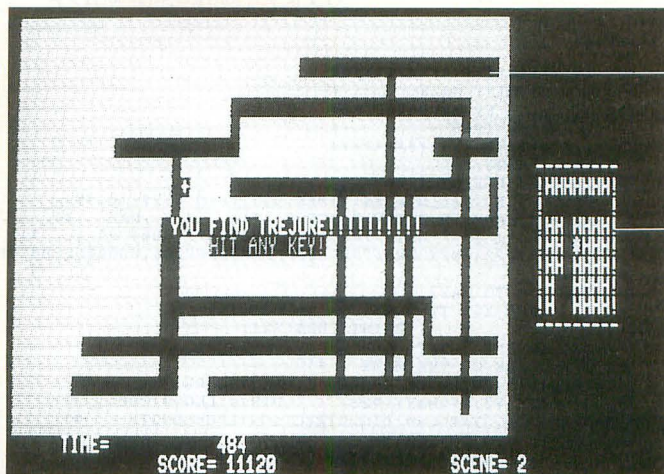
で，地図ですけど，左側のブロックとまったく同じ表現の仕方じゃ地図の意味がないから，なんかないかなーと思ったらありました。PCG(つまりCGEN 1)で描かれたものをSCRN$でとって普通のANK文字(CGEN 0)で書くとそのまんま地図っぽくなっちゃうんですね，大発見です。

実は私は，X1のBASICのSCRN$はPCGで描かれたキャラかそうでないかを判定してくれるもんだ,だからSCRN$を使って地図を作ったらそのまんま青いブロックが出るだろうとばかり思ってたんですよね。マニュアルはよく見なきゃいけません。そうそう，私はプレイヤーのキャラを動かすのにINKEY$(0)を使ったのですが，あまりに速く動き過ぎてゲームにならないという声があったのでINKEY$に変えてしまいました(1750，1751行)。どーもこのゲームはキャラがキャタキャタしてBASICくさくてやだなー，とおっしゃる方はどうぞ1750行に'を付けて1751行からは外しちゃってください。

教訓6　始めにマニュアルありき

皆さんもいろいろ考えてネ

そうそう，ちょっと言い訳など。実のところこのゲームはちょっと簡単すぎるような気がするんですよね。っていうのもブロックに穴を掘り終わったらキーが押されるのを待ちにいくので，そのときどの辺に宝があるのかじっくり考えられちゃうんですよね。かといって穴を掘っている間長々と待たせてるのに，いきなり始めちゃうのも悪いだろうしと……。キーを押してから地図を表示すればいいのかもしれませんが。

さて，あとはスコアとGAME OVERの条件ですかね。私はただなんとなーくTIME制(つまり，一定の時間以内に宝を見つけないとゲームオーバー。このタイムも面数が増えるごとにどんどんと減っていく）にしてしまったわけです。スコアのほうも宝を見つけた時点の残り時間×10という単純なものになっています。

Mini-Mini MAZE

じゃ，ひと通りゲームを説明しておきましょう。左側に迷路とプレイヤー"+"がいます。で，宝は画面右側の（迷路の一部を切り取った）地図の"＊"が指す場所にあります。地図の"H"は壁です。時間切れになるまでに頑張って宝を見つけてください。自分の移動は2・4・6・8キーで宝がありそうな場所でスペースを押してください。残り時間×10がスコアになります。時間切れでゲームオーバーです。

ちょっと考えたらこれ以外の方法はいくらでも見つけられるはずですよね。たとえば，時間じゃなくて移動のステップ数を条件にするとか，宝をたくさん埋めておき一定個以上見つけられたら1面クリアのオリエンテーリングゲームとか（その場合は青いブロックよりも緑のブロックのほうがいいかもしれない),敵が出てきてわやわやブロックを食っちゃうとか地図を左右逆にしちゃうとか，まあ，無限大に発散しそうなくらいにいろいろ出てくるわけです。

教訓7　星の数だけゲームネタはある。

最後に，パソコンゲームでなくて，いろいろ公園で遊ぶゲーム（っていうか子供の遊び）っていうのあるわけですよね。三角ベース缶蹴りとか泥警とか。みんなそういうものっていうのは公園というひとつのフィールドのなかでいろんなルールを作って遊んでしまうわけです。同じ場所でもルールを変えればまったく違うゲームを作ることもできるし，それは昔みんながやってきたことがあるはずのものですからね。誰もが面白い暇プロを作れる素質っていうのはあると思います。だから，しっかりと感性を磨いて小さなプログラムでもいいからゲームを作ってみてください。そして，Oh！X編集室へ送って原稿料を稼いじゃってください。

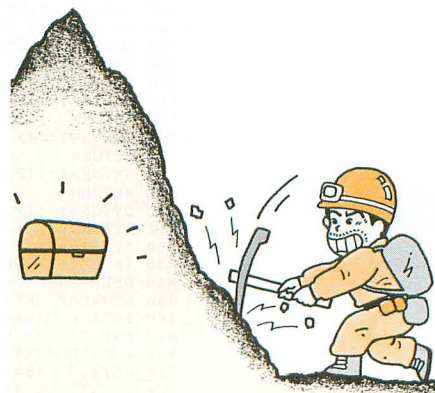
それでは，プログラムが完成したあかつきには，(で)の「オタッキー養成講座」係へぜひどうぞ（え，ウソですよ。できるわけないでしょ，そんなコーナー)。

教訓8　身からでたオタッキー

2)　知らないかもしれないけど，プロレスの伝統的ルールによると，5秒以内の反則は反則でないそうで，それがプロレスの試合をエンターテイメントとして盛り上げているというわけ。こーゆーことを許しちゃうのがBASICのいいところなんじゃないかな。

3)　アトリビュートというのはキャラクターの属性を決めるエリア。キャラクターの色とか，CGENいくつになってるかとかね。あ，知ってた？　すいません。

4)　最低っていえば，Oh！X お笑い3人組筆頭与力の金子一っ！　おまえ，いったい何考えてんだよー。看護婦さんがバッタバッタと病人をなぎ倒していくゲーム「患者ウォーリアーズ」，最後のボスキャラが○○○○だとー!?平成になったからできなくなってほんとーによかったと思うよ，私ゃ。

5)　再帰呼び出しっていうのはBASICでは使いにくいテクニックなんだとでも思っておいてください。CやPascalだったら簡単なんですけどね。

リスト2 Mini-Mini MAZE

```
10 '************************
20 '* MINI-MINI MAZE        *
30 '*         By Hu-BASIC   *
40 '*  Decno 1989           *
50 '************************
60 '(*   INIT; *)
70 WIDTH 80:DEFINT A-Z
80  RANDOMIZE (VAL(RIGHT$(TIME$,2))):DIM SCR$(10,10)
90 MAXX=50:MAXY=20:SCORE=0:SCE=0 :'(*   X,Y ﾉMAX   *)
100 WAIT0=10:'(*   WAIT ﾀｲﾑ(ﾀﾀﾞｼ 650ﾃﾞﾊﾅｸ 660ｷﾞｮｳﾉ)   *)
110 DEFCHR$(72)=HEXCHR$("FFFFFFFFFFFFFFFF0001600000288820 55AB75AA55AADDAA")
120 CGEN 1
130 FOR Y=0 TO MAXY
140      FOR X=0 TO MAXX
150                 PRINT CHR$(72);
160 NEXT:PRINT :NEXT:CGEN 0
170 '(*  MAKE UP THE MAZE ! *)
180 LOCATE15,22:PRINT"SCORE=";:PRINTSCORE
190 GX=MAXX ¥2:GY=MAXY ¥2:       DIR=0:LNG=0:ODD=1
200 '(*   GX,GY=ｻﾞﾋｮｳ:DIR=DIRECTION:LNG=LENGTH    *)
210 FOR I=1 TO 20-SEC+1
220    RANDOMIZE(VAL(RIGHT$(TIME$,2)))
230    DIR=INT(RND(1)*4)+1
240    IF (DIR=DIR2) OR ((DIR MOD 2)=(DIR2 MOD 2)) THEN 230
250    DIR2=DIR
260    LNG=INT(RND(1)*(INT(MAXX)))+2 :RANDOMIZE(VAL(RIGHT$(TIME$,2)))
270    DX=0:DY=0
280    IF DIR =5 THEN 230
290         ON DIR GOTO  300,310,320,330
300         DX=-1:DY=0  :GOTO 340
310         DX=0:DY=+1  :GOTO 340
320         DX=+1:DY=0   :GOTO 340
330         DX=0:DY=-1  :GOTO 340
340         (*  ON GOSUB"  ﾉ end;  *)
350         FOR J=1 TO LNG
360         IF GX+DY>MAXX-2 OR GX+DX<1 THEN GX=INT(RND(1)*MAXX+1 )
370         IF GY+DY>MAXY-2 OR GY+DY<1 THEN GY=INT(RND(1)*MAXY+1)
380         GX=GX+DX:GY=GY+DY:LOCATE GX,GY :PRINT" "
390 NEXT:NEXT
400 '(*      ﾀｶﾗｦｳﾒﾙ          *)
410 TX=INT(RND(1)*(MAXX-4))+4
420 TY=INT(RND(1)*(MAXY-4))+4
430 IF SCRN$(TX,TY,1)=" " THEN 400
440 'LOCATE TX,TY:PRINT"*" '(*   DEBUG ﾖｳ   *)
450 FOR VX=MAXX+3 TO MAXX+11
460       LOCATEVX+1,MAXY¥2-3:PRINT"-";:LOCATEVX+1,MAXY¥2+5:PRINT"-";
470 NEXT
480 FOR VY=MAXY¥2-2 TO MAXY¥2+4
490       LOCATEMAXX+4,VY:PRINT"!";:LOCATEMAXX+12,VY    :PRINT"!";
500 NEXT
510 FOR X=0 TO 6
520      FOR Y=0 TO 6
530           LOCATE MAXX+5 +X ,MAXY¥2+Y-2 :PRINT SCRN$(TX+X-3,TY+Y-3,1)
540 NEXT:NEXT
550 LOCATE MAXX+5 +3 ,MAXY¥2+1:PRINT"*"
560 '(*   ********* GAME MAIN***********   *)
570 BEEP:LOCATE5,21:PRINT"GAME START":LOCATE45,22:PRINT"SCENE="+STR$(SCE):A$=INK
EY$(1)
580 LOCATEGX,GY:PRINT"+"
590 BEEP:LOCATE5,21:PRINT"              "
600 'TIME ﾉ ｾｯﾃｲ
610 SCE=SCE+1:LT=1101-SCE*100:LOCATE 5,21:PRINT"TIME="
620 'GAME MAIN
630 LT=LT-1:LOCATE20,21:PRINTLT
640 IF LT=0 THEN 900
650 A$=INKEY$:LOCATE75,24:IF A$="" THEN 630
660 'FOR WAIT=0 TO WAIT0:KEY0,"":A$=INKEY$(0):LOCATE75,24:IF A$="" THEN 630:'(*
...ｷｰｶﾞﾆﾌﾞｸﾃｲﾔﾀﾞ!ﾄｲｳｶﾀﾊｺﾁﾗｦﾄﾞｳｿﾞ"*)
670 IF A$=CHR$(32) THEN 830
680 A=VAL(A$):IF A=0 THEN 630
690 LOCATE GX,GY:PRINT" ";
700 ON A GOSUB 730,740,730,760,730,780,730,800
710 LOCATE GX,GY:PRINT"+";
720 GOTO 630
730 RETURN
740 GY=GY+1 :IF GY>MAXY-1 THEN GY=MAXY-1
750 RETURN
760 GX=GX-1 :IF GX<1      THEN GX=1
770 RETURN
780 GX=GX+1 :IF GX>MAXX-1 THEN GX=MAXX-1
790 RETURN
800 GY=GY-1 :IF GY<1      THEN GY=1
810 RETURN
820 'ｱﾀﾘﾉｼｮﾘ!!!!!!!!!!!!!!!!!
830 IF GX<>TX OR GY<>TY THEN BEEP:GOTO 630
840 BEEP:BEEP:FOR A=0 TO  1000:BEEP1:NEXT:BEEP0
850 SCORE=SCORE+LT*10
860 LOCATE MAXX¥2-10,MAXY¥2
870 PRINT" YOU FIND TREJURE!!!!!!!!!!"
880 LOCATEMAXX¥2-5,MAXY¥2+1:COLOR2:PRINT"HIT ANY KEY!":COLOR7
890 KEY0,"":A$=INPUT$(1):CLS:GOTO 120
900 LOCATE TX,TY:PRINT"*"
910 LOCATE 0,21:PRINT"How stupid! Treasure is (*)!":KEY0,""  :END
```

なんでもありのプログラミング

「ただの双六」でたんばルンバ

BASIC特有のぎこちなぁいプログラムを連想しがちなあなた，このアップテンポなスゴロクはいかがでしょう。ぜひ自分なりのアイデアを考える材料にしてください。

Nagasawa Atsuhiro
長澤 淳博

ここは都内某所，X1turboのある部屋です。今日は，Oh！X編集部極秘BASIC作戦の開発現場に忍び込んでみました。

おおーっと，現場はすでにかなり混乱しています。散乱するフロッピーディスクに，とぐろを巻くプリンタ用紙。この世のものとは思えません。

しかし，我々は今，あるBASICプログラムを発見しました。その謎のディスクを調べると，そこにはあるファイルネームが刻まれていたのでした。

なんと，その正体は「ただの双六」，関西弁では「ただの双六」だったのです。

おや，どうやらプランナー兼プログラマの長澤氏がベンツで戻ってきたようです。さっそく，お話を伺ってみましょう。

まあ，企画ですねえ

なんだかゲームレビューみたいなノリになってしまいましたが，主旨はプログラミングのお話です。

プログラミングといえば，私などの場合，なんか作ろうかなと思ったら，たいてい市販のソフトやゲーセンとかにアイデアの原型を求めます。逆にいえば，ゲーセンでやりたいゲームを作ろうとするんですねえ。これが，そもそもの挫折の原因なのです。今回だって，特集用に基本的なゲームを1本ということだったのですが，頭の中はスペースハリアーのようなシューティングゲームとか，大戦略のような大掛かりなシミュレーションゲームとかのイメージに侵され，おもいっきり悩んでしまうわけです。

やはりBASICをおもちゃにしようとするなら，発想の転換が必要でしょう。プログラミングを楽しむためのポイントは，企画に一番時間をかけることではないでしょうか。まあ企画といっても，あれこれボーっと考えているだけだったりすることも多いのですが。ちなみに「現在，企画をつめているところです」などといって締め切り間近までプログラミングをしなくてもすむのはBASICのいいところです。

ここに紹介する「ただの双六」のプログラムは正味2日ぐらいの作業で仕上げられる程度のものとして用意しました。

私が作ったときは実際にどうだったかというと，「たんば」の広告とOh!X12月号の「ROGUEスゴロク」とを見てから，スゴロクを作ろうと思い立つまでに3日，画面構成を考えるのに1日，花火のルーチンを作って1日，そのほかの部分に2日といった感じです。

さいころマジック～♪

それでは，このゲームの内容について簡単に説明しておきましょう。とりあえず9人まで同時にプレイすることができますから3世代同居型住宅の皆さんでも一家だんらんの楽しいひとときが過ごせます。おっと，忘れるとこでしたがこのプログラムではコンピュータもプレイヤーとして参加するようになってますから，ひとりでもそれなりに楽しめるでしょう。X1マークのキャラクターがコンピュータくんのコマですので仲間に入れてやってください。

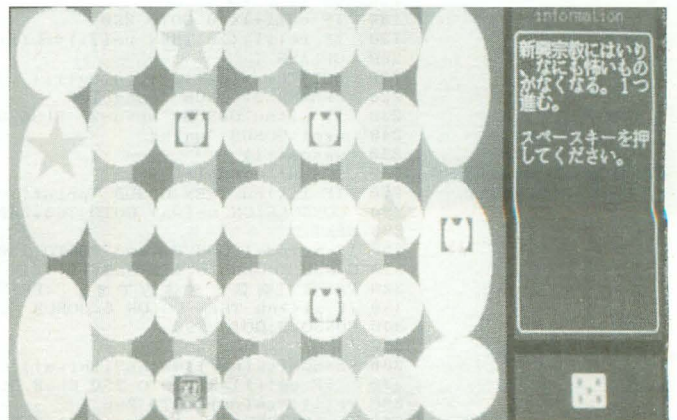
プログラムを実行させたら，何人でプレイするかを聞いてきますから人数と，それに続いてメンバーの名前をひとりずつ入力してください。あとは「○○○さんの番です」と表示されて，自動的にサイコロが回転しますからスペースキーを押して止めてください。これだけの操作でゲームが進行します。BASICとはいえ，なかなか軽快なテンポのゲームですよ。

この「ただの双六」のモットーは，簡単操作で誰にでも使用法がわかり，いくらでも改造可能であること，そして，これがいちばん大切なことなのですが，プログラムの開発者であるところの（あ，要するに私のことか）独断と偏見によって構成されていることでしょう。「こんなものどうするんだ！」との投書をいただいても，「なんでもありです」とお答えするしかありません。Oh!Xの読者たるもの，BASICのプログラミングとはそういうものだと悟らねばなりません。明日はアタシの風が吹き，昨日のプログラムは今日の敵なのです。

現代に生きるパソコンユーザーの皆さんはむやみに実用品を作ろうなんて思ってはいませんか？　いけません，いけません。むしろ，寝る前にその日のディスケットをフォーマットしてしまうくらいの覚悟で1日1日を精一杯生きていく，そんな流浪のプログラマの生き方にピッタリなのがBASICというわけです。

BASICだからなんでもあり

「言語としてのBASICはチンドン屋のようなものだ」と非難する言語オタクが多い今日このごろですが，「結局どんな手段を使おうが自分の好きなようにできればそれでいいじゃないか」という乱暴さが逆にBASICの魅力ではないでしょうか。

もちろん，この「ただの双六」のプログ

マス目ごとに恐ろしいメッセージが隠されている

ラムも「なんでもあり」になっています。もしPCGデータが面倒臭ければ，打ち込まないで自分で作ったり，変数をちょいちょいと変えてやったりすればいいのです。また，出てくる文章がつまらなかったら，自分で考えながら打ち込んでください。サイコロルーチンだけを取り出して自分のプログラムに応用したっていいですし，花火をもっと花火らしくすれば楽しさ倍増です。

ここまで書くと，いろいろとアイデアが出てきます。あるマス目に着いたら別のゲームが始まったり，ゲーム盤をスクロールさせて大きくし，人生ゲームのようにしたりもできるでしょう。地上げの王様なんかが移植されても面白そうです。

また，マシン語の知識を持ったうえでBASICを使えば，なかなか強力なことができるものです。たとえば，この「ただの双六」の画面に女の子の顔を出して口パクをやらせたいとします。グラフィックで絵を描くほどの絵心はありませんから，テレビから2枚の画像を取り込んで適当な大きさにします。それをメモリ上にでも置いておいて，DMA転送（『試験に出るX1』より）を使えばできるでしょう。言うはやすし，行うは……やっぱり簡単ですよね。頑張れば，イースのオープニングもどきだってできるだろうし，「たんば」を真似ればもう「死ぬのはぜんぜんこわくない」。

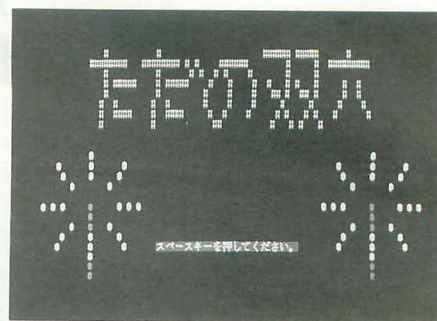
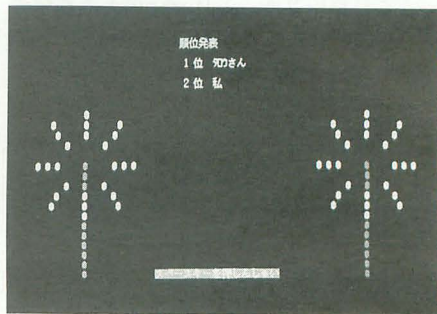
このように単なるスゴロクでも，読者の皆さんがいかなるゲームを連想するかによってこのプログラムの価値は変わってくるのです。作りかけて未完成のまま放置されたゲームなどを組み込むなりして原型をとどめないくらいメチャクチャにしてやってください。

プログラムの解説

プログラムはX1turbo（CZ-8FB02）用に書かれています。オールBASICですから，画面モードの設定やグラフィック命令を変更し，漢字やPCGの扱いに注意すれば他機種への移植は簡単でしょう。

X1では，30行のWIDTH 80,25,0,2をWIDTH 80に，40行のKLIST 0:KMODE 1を削除して漢字はカタカナに変更してください。

120～300行がタイトルを含むメインルーチンで，160～290行が人数分だけループする1ターンルーチンです。また，310～410行は進み戻しの処理，そのあとは全部名前どおりのサブルーチン群です。

マス目に順に0から番号をふり，メッセージや，いくつ進んだかなどの情報は配列にあらかじめ用意しておいて，プレイ中には読み出すだけです。配列で30個宣言してあるのは各マス目の情報，px，pyは花火用，宣言もせずに使っているのは各コマの情報です。なお，コマの絵はPCGで作って，これを配列km$に入れています（1320～1350行）。プレイヤーの人数は指定した数+1となっています（これはX1の分です）。

華麗なるオープニングシーンだ

エンディングには順位の発表もある

リスト1　ただの双六（X1turbo用）

```
10 '単なる双六だよ～ん              by  NAS          1989.1.12
20 '
30 CLS4:WIDTH 80,25,0,2       '高解像度専用
40 KLIST 0:KMODE 1:INIT:CLICK OFF
50 DIM xa(30),ya(30),me(30),ta(30),sh$(30),px(26),py(26),ax$(15)
60 LOCATE 28,12:PRINT "しばし、お待ちください。"
70 a=VAL(RIGHT$(TIME$,2)):FOR i=0 TO a*10:b=RND:NEXT
80 FOR i=0 TO 7:PALET i,0:NEXT
90 GOSUB "data":nu=1
100 GOSUB "graphic":GOSUB "chr"
110 '
120 '  MAIN 1
130 GOSUB "title"
140 FOR i=0 TO nu:re(i)=0:ed(i)=0:cs(i)=0:NEXT:ne=0
150 '
160 '  MAIN 2
170 FOR ii=0 TO nu
180  IF ed(ii)<>0 GOTO 290
190  IF re(ii)<>0 THEN re(ii)=re(ii)-1:GOTO 310
200  CFLASH 1
210  LOCATE xa(cs(ii)),ya(cs(ii)):ji=ii:GOSUB "km$"
220  CFLASH 0:GOSUB "console"
230  IF ii=nu GOSUB "space2" ELSE GOSUB "space1"
240  wz=1:GOSUB "walk"
250  wa=cs(ii)
260  GOSUB "console"
270  IF ii<>nu THEN GOSUB "print":PRINT:PRINT:COLOR 6:GOSUB "space0":COLOR 7
280  CONSOLE:ON me(wa) GOTO 360,380,390,400,410
290 NEXT
300 IF ne=nu+1 GOSUB "end":GOTO 130 ELSE GOTO 170
310 GOSUB "console":IF na$(ii)<>"" THEN PRINT na$(ii);"は、";
320 PRINT "病気でお休みです。":PRINT
330 IF ii<>nu THEN COLOR 6:GOSUB "space0":COLOR 7
340 CONSOLE:GOTO 290
350 '
360 wz=SGN(ta(wa)):w0=ABS(ta(wa)):GOSUB "walk"
370   IF cs(ii)=30 GOTO 250 ELSE GOTO 290
380 re(ii)=ta(wa):GOTO 290
390 wa=INT(RND*29+1):IF me(wa)=3 GOTO 390 ELSE GOTO 260
```

```
400 ne=ne+1:ed(ii)=ne:GOTO 290
410 wz=-1:w0=cs(ii):GOSUB "walk":GOTO 290
420 '
430 LABEL "km$"
440 CGEN 1:KMODE 0:PRINT km$(ji):KMODE 1:CGEN 0:RETURN
450 '
460 LABEL "print"
470 b$=sh$(wa)
480 FOR i=1 TO LEN(b$) STEP 2
490  c$=MID$(b$,i,2):PRINT c$;:FOR j0=0 TO 80:NEXT
500  IF c$="。" THEN FOR j0=0 TO 200:NEXT
510  SOUND 7,&B111110:SOUND 13,9:SOUND 11,0,10:SOUND 0,0,1:SOUND 8,16
520 NEXT:RETURN
530 '
540 LABEL "koma"   'in..w
550 FOR j1=0 TO nu:LOCATE xa(cs(j1)),ya(cs(j1))
560 ji=j1:GOSUB "km$":NEXT
570 RETURN
580 '
590 'in.. w0,wz(1,-1)
600 LABEL "walk"
610 w=cs(ii)
620 FOR jj=1 TO w0
630  LOCATE xa(w),ya(w):PRINT ku$
640  SOUND 7,&B111110:SOUND 13,9:SOUND 11,0,12:SOUND 0,0,2:SOUND 8,16
650  w=w+wz:LOCATE xa(w),ya(w):ji=ii:GOSUB "km$":FOR j0=0 TO 200:NEXT
660  IF w=30 OR w=0 THEN jj=w0
670 NEXT:cs(ii)=w:GOSUB "koma":RETURN
680 '
690 LABEL "space0"
700 PRINT "スペースキーを押してください。"
710 KEY0,"":CONSOLE:w0=RND
720 a$=INKEY$
730 IF a$=" " THEN RETURN ELSE GOTO 720
740 '
750 LABEL "space1"
760 IF na$(ii)<>"" THEN PRINT na$(ii);"の番です。"
770 KEY0,"":PRINT "サイコロをふります。":PRINT
780 COLOR 6:PRINT "スペースキーを押してください。":COLOR 7:CONSOLE
790 GOSUB 860:a$=INKEY$
800 IF a$=" " THEN RETURN ELSE GOTO 790
810 '
820 LABEL "space2"
830 PRINT "私の番です。":CONSOLE
840 a0=INT(RND*20+20):FOR i=0 TO a0:GOSUB 860:NEXT:RETURN
850 '
860 KMODE 0:w0=INT(RND*6+1)
870 SOUND 7,&B111110:SOUND 13,9:SOUND 11,0,3:SOUND 0,0,5:SOUND 8,16
880 CGEN 1:LOCATE 68,21:PRINT sa$(w0);:CGEN 0:KMODE 1:RETURN
890 '
900 LABEL "console"
910 CONSOLE 2,16,62,16:CLS:RETURN
920 '
930 LABEL "end"
940 CLS:FOR i=1 TO 7:PALET i,0:NEXT
950 COLOR 6:LOCATE 30,0:PRINT "順位発表":COLOR 7
960 FOR i=0 TO nu:LOCATE 30,ed(i)*2:PRINT ed(i);"位  ";na$(i);:NEXT
970 PAUSE 6
980 hx=10:hy=12:GOSUB "hanabi":hx=69:hy=12:GOSUB "hanabi"
990 LOCATE 25,23:CFLASH 1:PRINT "スペースキーを押しんしゃい";:CFLASH 0
1000 a$=INKEY$:IF a$=CHR$(&H1B) RETURN 1020
1010 IF a$=" " RETURN ELSE GOTO 1000
1020 END
1030 '
1040 LABEL "title"
1050 CONSOLE:CLS:FOR i=0 TO 7:PALET i,0:NEXT
1060 KMODE 1:LOCATE 0,1:COLOR 5:a=10
1070 PRINT SPC(a);"   #          #                      ###### #####      #    "
1080 PRINT SPC(a);"########   ######## # #    ######        #     #   #########"
1090 PRINT SPC(a);"   #          #      # #  # #    #       #     #            "
1100 PRINT SPC(a);"   # #####    # #####    #   #    #  #   # #   #     #  #   "
1110 PRINT SPC(a);"   #          #         #   #     #   ## #  ## #     #  #   "
1120 PRINT SPC(a);"  #  #       #  #       #   #     #     #     #     #    #  "
1130 PRINT SPC(a);"  #  ####    #  ####     #  #    #     # #   # #   #       #"
1140 PRINT SPC(a);"  #          #            ##          #   # #   #           "
1150 COLOR 7
1160 hx=10:hy=16:GOSUB "hanabi":hx=69:hy=16:GOSUB "hanabi"
1170 LOCATE 24,20:CFLASH 1:GOSUB "space0":CFLASH 0
1180 FOR i=0 TO 7:PALET i,i:NEXT
1190 CLS:COLOR 2:LOCATE 64,0:PRINT "information";:COLOR 7
1200 KMODE 0:LOCATE 61,1:PRINT CHR$(&H9E);STRING$(16,CHR$(&H90));CHR$(&H9B)
1210 FOR i=2 TO 17:LOCATE 61,i:PRINT CHR$(&H91);SPC(16);CHR$(&H91):NEXT
1220 LOCATE 61,18:PRINT CHR$(&H9C);STRING$(16,CHR$(&H90));CHR$(&H9D):KMODE 1
1230 GOSUB "console"
1240 PRINT:INPUT "何人でやりますか?(1-9)",nu:IF nu<1 OR nu>9 GOTO 1240
1250 na$(nu)="私"
1260 FOR i=0 TO nu-1:PRINT i+1;"番目の名前は?";:INPUT "",na$(i)
1270 IF na$(i)<>"" THEN na$(i)=na$(i)+"さん"
1280 NEXT:COLOR 6:b$=""
1290 INPUT "これでよいですか(y/n)",b$
1300 COLOR 7:IF b$="n" OR b$="N" GOTO 1240
1310 CLS:CONSOLE
```

```
1320 kw$=CHR$(&H1F)+STRING$(4,CHR$(&H1D))
1330 FOR i=0 TO nu-1
1340  km$(i)=CHR$+(&HE8,&HE9,&HEA,&HEB)+kw$+CHR$(&HF8,&HF9,&HFA,&HFB)
1350 NEXT:km$(nu)=CHR$(&HEC,&HED,&HEE,&HEF)+kw$+CHR$(&HFC,&HFD,&HFE,&HFF)
1360 FOR i=0 TO 15:ax$(i)="":NEXT
1370 FOR i=0 TO 15:FOR j=0 TO 3:ax$(i)=ax$(i)+CHR$(&HC0+i*4+j):NEXT:NEXT
1380 FOR i=1 TO 4:sa$(i)=ax$(i-1)+kw$+ax$(i+3):NEXT
1390 FOR i=5 TO 6:sa$(i)=ax$(i+3)+kw$+ax$(i+7):NEXT
1400 ku$="    "+kw$+"    ":RETURN
1410 '        グラフィック
1420 LABEL "graphic"
1430 FOR i=0 TO 479 STEP 80
1440  LINE (i,0)-(i+39,191),PSET,&H41,bf
1450  LINE (i+40,0)-(i+79,191),PSET,&H40,bf
1460 NEXT
1470 cx=&H67
1480 FOR i=79 TO 398 STEP 64
1490  IF cx=&H67 THEN cx=&H75 ELSE cx=&H67
1500  FOR j=0 TO 191 STEP 40
1510   CIRCLE@ (i+31,j+15),32,7,.5:PAINT (i+31,j+15),cx,7
1520  NEXT
1530 NEXT
1540 CIRCLE@ (47,20),40,7:CIRCLE@ (431,170),40,7
1550 CIRCLE@ (430,36),36,7,1.01:CIRCLE@ (430,115),36,7,1.01
1560 CIRCLE@ (47,76),36,7,1.01 :CIRCLE@ (47,156),36,7,1.01
1570 PAINT (47,20),7:PAINT (431,170),7
1580 PAINT (430,36),&H67:PAINT (430,115),&H67
1590 PAINT (47,76),&H67 :PAINT (47,156),&H67
1600 SYMBOL (24,8),"あがり",1,1,3,0,PSET
1610 SYMBOL (400,160),"ふりだし",1,1,3,0,PSET
1620 LINE (488,0)-(631,151),PSET,&H10,bf
1630 LINE (488,159)-(631,191),PSET,&H10,bf
1640 '
1650 FOR i=1 TO 29:IF me(i)=3 GOSUB 1670
1660 NEXT:RETURN
1670 e1=xa(i)*8+14:e2=ya(i)*8+8
1680 POLY (e1,e2),16,3,144,90,810:PAINT (e1,e2),3:PAINT (e1,e2-8),3
1690 PAINT (e1+9,e2-4),3:PAINT (e1-9,e2-4),3
1700 PAINT (e1+4,e2+6),3:PAINT (e1-4,e2+6),3:RETURN
1710 'データ
1720 LABEL "data"
1730 ww=1:j=44:w1=-8
1740 FOR i=21 TO 1 STEP -5
1750  FOR ii=1 TO 5
1760   xa(ww)=j:ya(ww)=i:ww=ww+1
1770   j=j+w1:IF j<12 OR j>44 THEN w1=-w1:j=j+w1
1780  NEXT
1790 ww=ww+1:NEXT
1800 xa(0)=52:xa(12)=52:xa(24)=52:ya(0)=21:ya(12)=12:ya(24)=2
1810 xa(6)=4:xa(18)=4:xa(30)=4:ya(6)=17:ya(18)=7:ya(30)=1
1820 RESTORE 2740
1830 FOR i=1 TO 30:READ me(i),ta(i),sh$(i):NEXT
1840 ww=0
1850 FOR i=3 TO 5:FOR j=0 TO 360 STEP 45
1860  w=RAD(j):px(ww)=COS(w)*i*2:py(ww)=SIN(w)*i:ww=ww+1
1870 NEXT:NEXT
1880 RETURN
1890 '花火
1900 LABEL "hanabi" 'in... hx,hy
1910 KMODE 0:COLOR 1
1920 SOUND 7,&B111110
1930 SOUND 8,16:SOUND 11,0, 30,9:ii=30
1940 FOR i=23 TO hy STEP -1
1950  LOCATE hx,i:PRINT CHR$(&HE0);
1960  SOUND 0,ii,1:ii=ii+1
1970  FOR w=0 TO 100:NEXT
1980 NEXT
1990 PAUSE 6
2000 SOUND 7,&B110110:SOUND 6,10:SOUND 0,0,2:ii=15:COLOR 6
2010 FOR i=0 TO  26
2020   SOUND 8,ii:ii=ii-1:IF ii<0 THEN ii=0
2030   LOCATE hx+px(i),hy+py(i):PRINT CHR$(&HE0);
2040 NEXT
2050 KMODE 1:COLOR 7
2060 RETURN
2070 '
2080 LABEL "chr"
2090 DEFCHR$(192)=HEXCHR$("7FFFFFFFFFFFFFFF7FFFFFFFFFFFFFFF7FFFFFFFFFFFFFFF")
2100 DEFCHR$(193)=HEXCHR$("FFFFFFFFFFFFF8F0FFFFFFFFFFFFFFFFFFFFFFFFFFFFF8F0")
2110 DEFCHR$(194)=HEXCHR$("FFFFFFFFFFFF1F0FFFFFFFFFFFFFFFFFFFFFFFFFFFFF1F0F")
2120 DEFCHR$(195)=HEXCHR$("FEFFFFFFFFFFFFFFFEFFFFFFFFFFFFFFFEFFFFFFFFFFFFFF")
2130 DEFCHR$(196)=HEXCHR$("7FFFFFFFFFFFFEFC7FFFFFFFFFFFFFFF7FFFFFFFFFFFFEFC")
2140 DEFCHR$(197)=HEXCHR$("FFFFFFFFFFFF1F0FFFFFFFFFFFFFFFFFFFFFFFFFFFFF1F0F")
2150 DEFCHR$(198)=HEXCHR$("FFFFFFFFFFFFF8F0FFFFFFFFFFFFFFFFFFFFFFFFFFFFF8F0")
2160 DEFCHR$(199)=HEXCHR$("FEFFFFFFFFFF7F3FFEFFFFFFFFFFFFFFFEFFFFFFFFFF7F3F")
2170 DEFCHR$(200)=HEXCHR$("7FFFFFF8F0F0F8FF7FFFFFFFFFFFFFFF7FFFFFF8F0F0F8FF")
2180 DEFCHR$(201)=HEXCHR$("FFFFFF7F3F3F7CF8FFFFFFFFFFFFFFFFFFFFFF7F3F3F7CF8")
2190 DEFCHR$(202)=HEXCHR$("FFFFFFFFFFFF3F1FFFFFFFFFFFFFFFFFFFFFFFFFFFFF3F1F")
2200 DEFCHR$(203)=HEXCHR$("FEFFFFFFFFFFFFFFFEFFFFFFFFFFFFFFFEFFFFFFFFFFFFFF")
2210 DEFCHR$(204)=HEXCHR$("7FFFF8F0F0F8FFFF7FFFFFFFFFFFFFFF7FFFF8F0F0F8FFFF")
2220 DEFCHR$(205)=HEXCHR$("FFFF7F3F3F7FFFFFFFFFFFFFFFFFFFFFFFFF7F3F3F7FFFFF")
2230 DEFCHR$(206)=HEXCHR$("FFFFFEFCFCFEFFFFFFFFFFFFFFFFFFFFFFFFFEFCFCFEFFFF")
```

▶１種情報処理受かりました。これもOh! Xのおかげです。けっこーいろいろなことが書いてあって役にたってます。そう，私に“不可”はあっても“不可能”はないっ！　先生一，単位くださーい。
岡田　忠宏（20）広島県

```
2240 DEFCHR$(207)=HEXCHR$("FEFF1F0F0F1FFFFFEFFFFFFFFFFFFFFFFEFF1F0F0F1FFFFF")
2250 DEFCHR$(208)=HEXCHR$("FFFFFFFFFFFFF7FFFFFFFFFFFFFFF7FFFFFFFFFFFFFFF7F")
2260 DEFCHR$(209)=HEXCHR$("F0F8FFFFFFFFFFFFFFFFFFFFFFFFFFFF0F8FFFFFFFFFFFF")
2270 DEFCHR$(210)=HEXCHR$("0F1FFFFFFFFFFFFFFFFFFFFFFFFFFFFF0F1FFFFFFFFFFFF")
2280 DEFCHR$(211)=HEXCHR$("FFFFFFFFFFFFFFFEFFFFFFFFFFFFFFFEFFFFFFFFFFFFFFFE")
2290 DEFCHR$(212)=HEXCHR$("FCFEFFFFFFFFFF7FFFFFFFFFFFFFFF7FFCFEFFFFFFFFFF7F")
2300 DEFCHR$(213)=HEXCHR$("0F1FFFFFFFFFFFFFFFFFFFFFFFFFFFFF0F1FFFFFFFFFFFF")
2310 DEFCHR$(214)=HEXCHR$("F0F8FFFFFFFFFFFFFFFFFFFFFFFFFFFF0F8FFFFFFFFFFFF")
2320 DEFCHR$(215)=HEXCHR$("3F7FFFFFFFFFFFFEFFFFFFFFFFFFFFFE3F7FFFFFFFFFFFFE")
2330 DEFCHR$(216)=HEXCHR$("FFFFFFFFFFFFF7FFFFFFFFFFFFFFF7FFFFFFFFFFFFFFF7F")
2340 DEFCHR$(217)=HEXCHR$("F8FCFFFFFFFFFFFFFFFFFFFFFFFFFFFF8FCFFFFFFFFFFFF")
2350 DEFCHR$(218)=HEXCHR$("1F3EFCFCFEFFFFFFFFFFFFFFFFFFFFFF1F3EFCFCFEFFFFFF")
2360 DEFCHR$(219)=HEXCHR$("FF1F0F0F1FFFFFFEFFFFFFFFFFFFFFFEFF1F0F0F1FFFFFFE")
2370 DEFCHR$(220)=HEXCHR$("FFFFF8F0F0F8FF7FFFFFFFFFFFFFFF7FFFFFF8F0F0F8FF7F")
2380 DEFCHR$(221)=HEXCHR$("FFFF7F3F3F7FFFFFFFFFFFFFFFFFFFFFFFF7F3F3F7FFFFF")
2390 DEFCHR$(222)=HEXCHR$("FFFFFEFCFCFEFFFFFFFFFFFFFFFFFFFFFFFFEFCFCFEFFFF")
2400 DEFCHR$(223)=HEXCHR$("FFFF1F0F0F1FFFFEFFFFFFFFFFFFFFFEFFFF1F0F0F1FFFFE")
2410 DEFCHR$(224)=HEXCHR$("7FFFF8F0F0F8FFFF7FFFFFFFFFFFFFFF7FFFF8F0F0F8FFFF")
2420 DEFCHR$(225)=HEXCHR$("FFFF7F3F3F7FFCF8FFFFFFFFFFFFFFFFFFFF7F3F3F7FFCF8")
2430 DEFCHR$(226)=HEXCHR$("FFFFFEFCFCFE3F1FFFFFFFFFFFFFFFFFFFFFEFCFCFE3F1F")
2440 DEFCHR$(227)=HEXCHR$("FEFF1F0F0F1FFFFFEFFFFFFFFFFFFFFFEFF1F0F0F1FFFFF")
2450 DEFCHR$(228)=HEXCHR$("7FF8F0F0F8FFF8F07FFFFFFFFFFFFFFF7FF8F0F0F8FFF8F0")
2460 DEFCHR$(229)=HEXCHR$("FF7F3F3F7FFF7F3FFFFFFFFFFFFFFFFFFF7F3F3F7FFF7F3F")
2470 DEFCHR$(230)=HEXCHR$("FFFEFCFCFEFFFEFCFFFFFFFFFFFFFFFFFFFEFCFCFEFFFEFC")
2480 DEFCHR$(231)=HEXCHR$("FE1F0F0F1FFF1F0FFEFFFFFFFFFFFFFFFE1F0F0F1FFF1F0F")
2490 DEFCHR$(232)=HEXCHR$("FFFEFDFBF7F7F6F5000103070F0F0F0F000103070F0F0F0F")
2500 DEFCHR$(233)=HEXCHR$("3FDFEFF7EA55AA55C0E0F0F8FFFFFFFFC0E0F0F8FFFFFFFF")
2510 DEFCHR$(234)=HEXCHR$("FCFBF7EF57AA55AA03070F1FFFFFFFFF03070F1FFFFFFFFF")
2520 DEFCHR$(235)=HEXCHR$("FF7FBFDFEFEF6FAF0080C0E0F0F0F0F00080C0E0F0F0F0F0")
2530 DEFCHR$(236)=HEXCHR$("FFFFFEFEFEFEFEFE0000010101010101000000000000000000")
2540 DEFCHR$(237)=HEXCHR$("FF00FFFFFFFFFFFF00FF000000000000000000000462C1838")
2550 DEFCHR$(238)=HEXCHR$("FF00FFFFFFFFFFFF00FF00000000000000000003C181830")
2560 DEFCHR$(239)=HEXCHR$("FFFF7F7F7F7F7F7F000080808080808000000000000000000")
2570 DEFCHR$(240)=HEXCHR$("FFFFF8F0F0F8FF7FFFFFFFFFFFFFFF7FFFFFF8F0F0F8FF7F")
2580 DEFCHR$(241)=HEXCHR$("F8FC7F3F3F7FFFFFFFFFFFFFFFFFFFFFF8FC7F3F3F7FFFFF")
2590 DEFCHR$(242)=HEXCHR$("1F3FFEFCFCFEFFFFFFFFFFFFFFFFFFFFF1F3FFEFCFCFEFFFF")
2600 DEFCHR$(243)=HEXCHR$("FFFF1F0F0F1FFFFEFFFFFFFFFFFFFFFEFFFF1F0F0F1FFFFE")
2610 DEFCHR$(244)=HEXCHR$("F0F8FFF8F0F0F87FFFFFFFFFFFFFFF7FF0F8FFF8F0F0F87F")
2620 DEFCHR$(245)=HEXCHR$("3F7FFF7F3F3F7FFFFFFFFFFFFFFFFFFF3F7FFF7F3F3F7FFF")
2630 DEFCHR$(246)=HEXCHR$("FCFEFFFEFCFCFEFFFFFFFFFFFFFFFFFFFCFEFFFEFCFCFEFF")
2640 DEFCHR$(247)=HEXCHR$("0F1FFF1F0F0F1FFEFFFFFFFFFFFFFFFE0F1FFF1F0F0F1FFE")
2650 DEFCHR$(248)=HEXCHR$("FAF5FAFDFAFFFFFF0F0F0F07070100000F0F0F0707010000")
2660 DEFCHR$(249)=HEXCHR$("AA55AA5D9A45A16FFFFFFFE7FFCF3703FFFFFFE7FFFF7E90")
2670 DEFCHR$(250)=HEXCHR$("55AA55BA59A285F6FFFFFFE7FFF3ECC0FFFFFFE7FFFF7E09")
2680 DEFCHR$(251)=HEXCHR$("5FAF5FBF5FFFFFFF0F0F0E0E080000F0F0F0E0E0800000")
2690 DEFCHR$(252)=HEXCHR$("FEFEFFFFF8F9F8F80101000007060707000000000000000000")
2700 DEFCHR$(253)=HEXCHR$("FF0000FF00FE0C0000FFFF3FFF0DFFFF64000000000C0C00")
2710 DEFCHR$(254)=HEXCHR$("FF0000FF007F030000FFFFCFF83FFFF30000000000030300")
2720 DEFCHR$(255)=HEXCHR$("7F7FFFFF1F9F1F1F80800000E060E0E0000000000000000000")
2730 RETURN
2740 '
2750 '1..進み、もどる 2..休み 3..スペシャル 4..ゴール 5..ふりだしへ
2760 DATA 1,-1,上野に行くと、自衛隊に勧誘されて入隊する。北の守りについて、1つもどる。
2770 DATA 1,2,今日は学校があるので、素直に行かないことにする。2つ進む。
2780 DATA 2,1,新宿にでようとして、間違えてOh!X編集部に行ってしまう。仕事を頼まれて、1回休み。
2790 DATA 3,0,ここで、すべしゃる。
2800 DATA 5,0,靖国神社付近で祝一平氏らしき人を発見する。氏の教えに感動して、ふりだしにもどる。
2810 DATA 0,0,渋谷に向かう。愛国党の演説は、まことにすばらしい。
2820 DATA 1,-2,新しいソフトを買ってたので急いで家に帰る。2つもどる。
2830 DATA 3,0,ここで、スペシャル。
2840 DATA 2,2,町でナンパしようとして、おもいっきりフラれる。悲しくて2回休み。
2850 DATA 1,1,六本木に行く。場違いなので、秋葉原に行く。安心して、1つ進む。
2860 DATA 1,-1,たまには学校に行ってみる。大学は慎重に選ぶべきだったと悟って、1つもどる。
2870 DATA 2,1,修学旅行に行く。夜になるまで身を潜めている。1回休み。
2880 DATA 3,0,ここで、SPECIAL.
2890 DATA 1,-3,修学旅行の夜、旅館のなかをウロウロして先生につかまる。トイレだと大嘘をついて、3つもどる。
2900 DATA 1,2,なんだかんだ言いながら、受験生になってしまった。とにかく勉強を始めて、2つ進む。
2910 DATA 2,1,予備校の模試をうけて、偏差値をみて気を失なって、1回休み。
2920 DATA 1,1,共通1次をうけて、700点取る夢をみる。幸先よく1つ進む。
2930 DATA 3,0,ここで、スペハリ。
2940 DATA 1,-2,夏休みになる。夏期講習にもいかずに、遊んで2つもどる。
2950 DATA 2,1,人生について迷う。その時こそ、Oh!Xを読んで、新たな境地を開くために、1回休み。
2960 DATA 1,-2,大学の先輩(友達)から電話がかかってきて、2つもどる。
2970 DATA 1,1,新興宗教にはいり、なにも怖いものがなくなる。1つ進む。
2980 DATA 2,1,受験で東京にでてきて、ついに消息不明になってしまい、1回休み。
2990 DATA 1,2,彼(彼女)とホイホイ遊ぶ。自分は今楽しければいいので、2つ進む。
3000 DATA 2,1,「受験生に正月はない」という講師の言葉を信じて、結局体調をくずしてしまい、1回休み。
3010 DATA 1,-3,せめて最後の神頼みに、亀戸天神に行く。これで合格なら苦労しないと思っていたら天罰で3つもどる。
3020 DATA 2,2,いきなり、平成の世になる。自粛して2回休む。
3030 DATA 3,0,ここいらで、スペシャル。
3040 DATA 5,0,代ゼミの申込書に記入をする。さあ、もう1年だ!ふりだしへ・・・
3050 DATA 4,0,晴れて、ゴールにたどりついた君は、えらい!
```

▶うおー！　このままでは最後の共通一次受験，じゃない受験生から第1回新テスト受験生になってしまうううーっ。最後の手段として二次の論文で祝文体を用いる計画である。
田爪　宏二(18)　宮崎県

ブロックテニスで反則攻撃

ちなみに2人で遊べるモードあり

マシン語による反則技の連続攻撃で出来上がったのが，このブロックテニス（X1用）です。これでも，もともとは純粋なBASICプログラムだったのだそーです。

Nishikawa Zenji
西川 善司

こんにちは，西川善司です。さっそくですが，皆さんはパソコンをなんのために購入したのでしょうか。ちょっとアンケートを取ってみましょう。

はい，それでは，パソコンをゲームのために購入した人，足で耳をほじってみてください。

あれーっ，結構少ないんだなぁ。ま，いいか。

私の場合は，もともとゲームがしたくてパソコン（MZ-700）を買ったのです。初めの2，3カ月は本当にゲームに明け暮れ，「カンニング大作戦」とか「ゾンビパニック」とか「スペースブラスター」とかいろいろ遊びました。やがて，自分でゲームを作ってみたいなんてことを思い始めまして，MZといっしょに付いてきたHuBASICをいじり始めたのでした。そして初めて完成したのが「数当てゲーム」。コンピュータが乱数で作った数を自分が当てるというやつです。

BASICの力は〜っ!!

はっきりいってBASICという言語はなかなかの曲者で，一見なんでもできそうで，実はなんにもできないというなんとも困った言語です。しかし，「ちょっと〜してみたい」ということになると，とんでもないくらいの威力を発揮します。

たとえば夜遅く，あなたは机に向かって漢字の書き取り練習をしていたとしましょう。ところが，ふと，突然，素数を求めたくなってしまいました。よくあることですね。このとき，隣の机に置いてあるパソコンの電源を入れ，いきなりアセンブラのソースを書こうとエディタを起動する人はあまりいないはずです。ほとんどの人がBASICを起動することでしょう。

では，BASICでゲームを作ることは可能でしょうか？　答えはもちろんOKです。「アフターバーナー」は無理だと思いますが，「平安京エイリアン」くらいならできそうです。動きのそんなに派手でない，いわゆるピコピコゲームならば，BASICで作ることが可能でしょう。このピコピコゲームを作ることはBASICを学習するという意味においても重要なことです。

ゲームというのは，さまざまな処理を必要とするので，先ほど言った「数当てゲーム」にしても，自分ひとりで作ったというのならBASICの半分以上を理解したといってもよいでしょう。これを読んでいる人のなかでも初心者の皆さんは，さっそく数当てゲームを作ってみましょう。もし，マニュアルを見ずに作れるようであれば，もう中級者への道が開けたことになると思います。

ゲーム作りはBASICを習得するための近道といえ，また，BASICでゲームを作るということは，今後もっと複雑なゲームを作るうえでの基礎になるでしょう。

ランク別BASICマスター法

ところで，私はパソコンユーザーを以下のようにランクづけしてみました。

はなもげら：

BASICはまったくわからない。そして覚える気もない。

初心者：

BASICは少し，または，ほとんど理解できる。マシン語はわからないが，覚える気はある（ここが重要）。

中級者：

マシン語が少しはできる。でも，オールマシン語はちょっとぉ……。

上級者：

アセンブラでオールマシン語のプログラムが書ける。

異常者：

16進コードでプログラムが書けて，コンピュータと直接コミュニケートができる。

というわけで，このランク別にBASICを習得するためのアドバイスをしていくことにします。

初心者は打ち込むべし

STEP 1：

まず，ランク「はなもげら」の人は経験値を稼いで早く「初心者」になってください。

そして初心者の方はひたすら打ち込みましょう。雑誌やマニュアルなどのプログラムをです。『マイコンBASIC Magazine』やOh!Xにはたくさんのプログラムリストが載っていますが，これを面倒臭がらずに自分の手でパソコンに打ち込むよう心がけましょう。別にプログラムを1行1行理解しなくてもかまいません。他人のプログラムを理解するのは結構困難なことですからね。

ただ，見たこともない命令があったら，必ずマニュアルで調べてみましょう。そして，入力が終わり，そのプログラムを遊んだり使ったりしてから，必ず，改造しましょう。ゲームなら設定（自機の数やスピードなど）を変えてみたり，ツールなら機能を拡張してみたり……。

STEP 2：

それができるようになったら，他機種のプログラムを自分のマシンに移植してみましょう。MZ-700ユーザーならPC-8001やパソピアなどのプログラムを，X1turboユーザーならPC-8801/9801やFM-7などのプログラムを移植してみましょう。

STEP 3：

そろそろ，BASICで何か自分で作ってみましょう。初心者でも作りやすいのは，パターンエディタ（PCGエディタ），迷路型ゲーム，サウンドエディタ，テキストアド

ベンチャー，RPGなどです。スピードを気にしないようなツール関係が比較的作りやすいようです。

STEP 4：

ボツを恐れず，Oh!Xに投稿しましょう。STEP4まで来られた人は，あれっ？　いつの間にか中級者になっていたとさ（ざーとらしい）。

中級者はBASIC＋マシン語すべし

ある程度パソコンをやっていれば，ここまでは来られます。が，パソコンに限らず，何をするにも中級者からが大変。このランクの人は，ひたすらBASIC＋マシン語のプログラムを書くようにしましょう。

とはいえ，自分でゼロから組むのが面倒な人は，雑誌に載っているプログラムなどの遅いところをマシン語にしてやりましょう。これを繰り返していれば，そのうちマシン語でプログラムを組むのがそれほど苦でなくなります。

上級者や異常者でBASICを知らない人はいないと思うし，仮にいたとしても，そういう連中は解析とかして覚えちゃうだろうから（うそうそ！）放っておいて，サンプルプログラムの説明にいくのでした。

BASICのはずが……

さて，いろいろ，忙しくて時間がなくて(強調表現)，プログラムを締め切り1日前から作り始めて，まあとにかくプロトタイプ（試作版）は完成したものの遅くて遊べない。おかしい。確か数年前，MZ-700で作ったときはオールBASICで遊べるものができたはずなのに……。

きっとX1のHuBASICはMZのより遅いに違いない，なんてことをブツブツいいながら，ボールの移動だけマシン語にしてみた。あれ？　まだ遅いなぁ。で，キー入力，ジョイスティック，ラケット移動，当たり判定，と次々にマシン語にしていったら，結局締め切りに間に合わなくなって編集室に泊まり込むはめになってしまいました。

んで，完成したのが今回のプログラムです。X1のHuBASIC用で，X1turboの人でもCZ-8FB01を使ってください（turbo BASICは不可)。マシン語部分（リスト2）はモニタコマンドやマシン語入力ツールなどから入力してください。BASICのモニタから打ち込んだ場合は，

SAVE“ファイル名”，&HFA00，&HFEFF，&HFA00

としてセーブしてください。

BASIC部（リスト1）も入力し終えたら，

CLEAR　&HFA00

としてから，先ほど入力したマシン語部分を，

LOADM“ファイル名”

でロードし，RUNしてください。PCGの設定で数十秒待たされます。

これは，中級者向けの「BASIC＋マシン語」のプログラムです。が，初心者の方も恐れることなかれ。画面まわりをいじったり，音楽を入れたり，手のつけられるところからどんどん改造していってみてください。

手始めに，簡単な部分から

それでは，プログラムの説明をします。

まず10行は，よくある画面の初期化（イニシャライズ)。650行にサブルーチンコールしているのはPCGをセットしているためです。

20行ではCLEAR文でマシン語エリアを確保し，マシン語ルーチンが読み込まれているかをチェック。30行ではUSR文の定義(後述)。70行でプレイヤーの数を聞いてきます。それで，80行でその値をワークにセット。同じように90〜100行でコンピュータのレベルをセットしています。

120行ではスコア用の変数をイニシャライズ。また，SPD＝&HE00とありますが，これはゲーム全体のスピードです。この値を小さくするとスピードが速くなり（1が最小値)，大きくすると遅くなります。130行については後述。140〜170行では，乱数によってボールが左右どちらから飛び出すか，また飛び出す方向を決定しています。180行は130行と同じく後述。190行では10を書き込んでいますが，これはラケットのY座標の初期値です。ラケットは2人分あるので，2回書き込んでいます。

220〜300行では，画面のレイアウトをしています。画面を見てデザインが気に入らない人はここを好きなように変えるだけでずいぶん違ったイメージのゲームになるでしょう(ちなみに私もまだ満足していない)。そして，310行から590行に飛んでいます。これはブロックの表示です。面数が1面しかないので，これまた気に入らない人は直してください。今の状態では7色のブロックが横1列に並ぶだけですので，ハート型とか星型とかに並べ換えてください。

320行で，プログラムの実行後すぐゲームが始まってはびっくりするので，何かキー入力があるまでループするようになっています。

USR関数でマシン語ルーチンへ

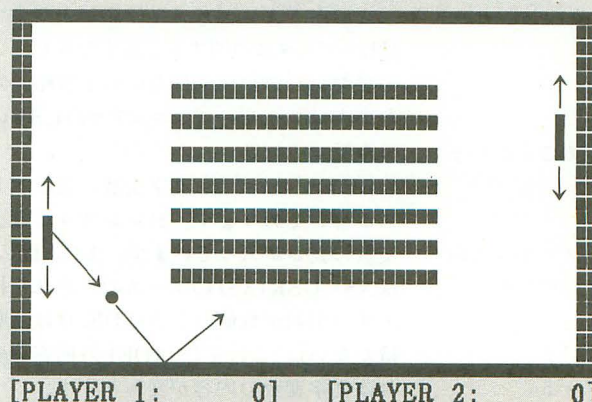
350，360行がメインルーチンです。

ここで，USRという命令が出てきました。これは，BASIC＋マシン語のプログラムを作るうえでとても大切な命令です。

一般にマシン語ルーチンにパラメータを渡す方法としては，130行や180行にあるような，MEM$やPOKEでメモリに書き込んで渡す方法があります。POKEは，

POKE adr,n

ブロックテニスゲーム「MAD BALL CLUB DX」

ゲームのルールは簡単。お互いにボールを打ち合って，相手のコートの中に入れれば勝ち。ただし，フィールド上のスペースウォール(ブロック)が，ボールの動きを複雑にするぞ。ラケットは，プレイヤー1がキーボード，プレイヤー2はジョイスティックで操作せよ。凶悪なコンピュータX1と対戦するモードあり，コンピュータ同士の対決も見られるモードあり。泣かせるなぁ。さあ，戦士よ！　X1を倒すのだ。では健闘を祈る（これじゃファミコンソフトの説明書だ)。

で，アドレスadrにnを書き込みます。また，MEM$は，

MEM$(adr,ln)=str

で，アドレスadrからlnバイト分strを書き込みます。

このstrにあたる部分は単なる文字変数だったりしますが，今回はMKI$というのを使っていますね。このMKI$というのはなかなか便利で，数式をバイナリに対応した文字列に変換します。たとえば，&HABCDは「へォ」となります。「へ」のASCIIコードは&HABですよね。そこで，

MEM$(adr,2)=MKI$(&HABCD)

とすると，adr+0番地にCDHが，adr+1番地にABHが書き込まれるわけです。

ABCDHがCD，ABの順になるのは，マシン語でLD (adr)，HLを実行したとき，Lがadr+0へ，Hがadr+1に格納されることと同じです。

で，LD HL，(adr)とすると，Lにはadr+0番地の内容が，Hにはadr+1番地の内容が書き込まれますよね。MEM$(adr,2)=MKI$(n)はちょうどZ80のマシン語の16ビットレジスタ(2バイトレジスタ)をメモリにしまうのと同じことです。

ところで，USRはソ連のことではありません。似たような命令にCALL文がありますが，CALL文では単にそのマシン語ルーチンにジャンプするだけです。一方，USRではメインルーチンからマシン語ルーチンにパラメータの受け渡しが可能なのです。

使い方ですが，まず，

DEFUSRn=adr

で関数定義を行います。nは0～9の番号，adrはマシン語ルーチンのエントリアドレスです。

実際に使うときは，

A=USRn(A)

のようにします。もちろんnは先ほどのDEFUSRnで定義した番号に対応します。0は省略可能です。

また，

A$=USR(A$)

という形式もあります。

2つの場合を分けて説明することにしましょう。

●X=USR(X)

まず，X=USR(X)のケースです。この場合，Aレジスタには変数Xの形式が入っており，

整数型　：　Aレジスタ=2
単精度型：　Aレジスタ=5
倍精度型：　Aレジスタ=8

となります。

HLレジスタには変数の格納されているアドレスが入っています。また，整数型のときには2バイトで[LOW][HIGH]の順に入っています。先ほどの&HABCDならば，CDH，ABHの順に入っているわけです。単精度のときも5バイトで指数1バイト，仮数4バイトの順で入っています。また，倍精度では8バイトで，指数1バイト，仮数7バイトの順で入っています。

Aレジスタは，その変数の形式が何バイト長かを表しているわけです。まあ，実用上よく使うのは整数型でしょう。

ところで，マシン語ルーチンからBASICに戻るときにこの(HL)の内容を変えたらどうなるのでしょう。X=USR(X)だったら変数Xの内容はもちろん変わります。今回はこの方法を使っています。

BASICリストの360行を見てください。「ON A GOTO」がありますね。これはマシン語ルーチン内で，もしバリケードに当たったら変数Aを1に，もしブロックに当たったら変数Aを2に，ゲームオーバーなら3にしており，BASICのメインプログラムに帰ったときにどんな仕事をしたらよいかを教えるための目印なわけです。

ですから，今回のサンプルでは変数Aは1から3までの値しか取りません。370行にはENDがありますが，これは縁起ものであって絶対ここに来ることはないのです。

BASICでの処理が終わって再びマシン語ルーチンに来たときはどうでしょうか。これはマシン語ルーチンのソースプログラム(リスト3)を見るとおわかりいただけると思いますが，プログラムの先頭で変数Aの値によってマシン語版の「ON A GOTO」処理を行ってますね（ソースリストの10～24行)。

●X$=USR(X$)

次に，X$=USR(X$)の場合はどうでしょうか。

このときAレジスタには3が入ります。HLレジスタは説明すると長くなるし，ここではたいして重要ではないので省略。知りたい人はユーザーズマニュアルのUSRの部分を見てください。

DEレジスタには，文字変数の格納アドレスが書いてあります。Bレジスタには文字変数の長さが入っています。また，DEとBはX$=USR(X$)のケースのみ意味を持ちます（引数が数値のときはDE，Bは意味を持たない)。これまた，(DE)の内容を変えれば文字変数の内容が変わります。

ところで，これまでの話では引数とUSRの持ち帰るべき変数の名前がいっしょでしたが，もちろん，

A=USR(B)
X$=USR(Y$)

のようなケースも大丈夫です（念のため)。

一般的に文字形式のほうが多くのパラメータを渡せるのでよく使われるようです。

再びBASICへ

さて，BASICプログラムの説明に戻ります。380～470行はゲームオーバー処理です。もしワンミスでゲームオーバーがイヤなら，ボールやラケットの座標を150～190行のように初期化してやって変数Aにゼロを入れて350行に飛ばしてやりましょう。また，1回ミスしたら中心のブロックの並びを変えてやるのもいいかもしれません。いわゆる1面クリアということです。

また，今回は使っていませんが，マシン語のソースリストのBREAK(FEEEH)というところにボールがいくつのブロックを崩したかを教えてくれるワークがあります。皆さんそれぞれで役立ててください（ブロックが全部崩れたら1面クリアとかね)。

500～570行はスコアを表示しているところです。初心者の人で，PRINT USINGやPRINT #n，を知らない人はマニュアルで確認しましょう。

580～630行はブロックを描くところ。先ほど書きましたが，ここにもっと凝れば，ゲームがもっと楽しくなるかもしれませんね。また，COLOR 0 つまり，黒のブロックを作るとこれは透明のブロックになります。また，ラケットと同じキャラクタ87H「■」を書いてやれば破壊不能のブロックとなるわけです。

640行から最後までがPCG定義ルーチンです。特に650～680行はROMCGパターンから太い文字（アタリ文字）を作り出すルーチンなので（結構有名だが）知らない人は覚えておきましょう。

マシン語ルーチンのソースリスト

リスト3のソースプログラムのほうも簡単に説明しておきましょう。

まず，10～24行。ここはUSR文の引数の値によって各サブルーチンへジャンプしています。

その後のLOOPというのがメインループで，まず何人でゲームをプレイしているかをチェック。もし0ならば，2人分のプレイをコンピュータが受け持つため，30行からHUMAN1というラベルの前までが，コ

ンピュータの移動ルーチンです。

アルゴリズムは単純で，ボールが一定距離より近づいてきたらラケットをボールの方向に動かしてやるというものです。実はこの一定距離というのがコンピュータのレベルで，この値が小さいほどミスをしやすくなるのです。

一方，人間がプレイヤーを受け持っている場合はキー入力ルーチンへサブルーチンコールしています。その後はプレイヤー2の処理で，同じく人間がプレイヤーでなければさっきと同じアルゴリズムでコンピュータ側のラケットを動かし，人間ならばジョイスティックの入力ルーチンへサブルーチンコールしています。

それから，ボールの動きを管理するルーチンにやって来ます。もし，ボールが右端もしくは左端ならば，ゲームオーバーなのでUSRの引数に3を入れてBASICに帰還します。

基本的にはボールの現在位置にあるキャラクタと次に当たるキャラクタをテキストVRAMから読み出して，今ボールが何に当たったのか，次はどこに当たるのかをチェックしています。もし，当たっているならボールの移動方向の処理後，USRの引数に，バリケードに当たったなら1を，ブロックに当たったのなら2を入れてBASICに帰還しています。

この処理後バリケードのスクロール，ゲーム全体のウエイトなどの処理をして再びメインループである，ラベルLOOPにジャンプしています。

その後は子ルーチンとワークエリアです。SE1～3は効果音です。ラベルWPSGでPSGのレジスタをセットし音を出しています。

＊　＊　＊

というわけで，今月はずいぶん慌ただしい内容でした(冗談も少なかったし)。久しぶりに「TRY AGAIN？」とか「GAME OVER」なんていう英語をプログラムに書いた西川善司でした。あ，朝になってる。今日は家に帰るぞ。

リスト1　メインプログラム(X1/turbo, CZ-8CB01/8FB01)

```
10 WIDTH 80:DEFINT A-Z:INIT:CLS4:GOSUB650
20 CLEAR &HFA00:IF PEEK(&HFA00)<>34 THEN END
30 DEFUSR0=&HFA00:KBUF OFF
40 '///// ESTABLISHING /////
50 L$=STRING$(3,HEXCHR$("871F1D"))
60 PALET 1,6:COLOR 7:CGEN1
70 INPUT "HOW MANY PLAYERS ?:",HM:IF HM>2 OR HM<0 GOTO70
80 POKE &HFEEC,HM
90 IF HM<2 THEN INPUT"COMPUTER LEVEL:",LV:LV=LV+2:IF LV>=256 GOTO90
100 POKE &HFEED,LV
110 CGEN:CLS
120 SC1#=0:SC2#=0:SPD=&HE00
130 MEM$(&HFEEA,2)=MKI$(SPD)
140 ON INT(RND(1)*2)+1 GOTO150,160
150 X=73:EX=-1:GOTO170
160 X=6:EX=1
170 EY=INT(RND(1)*3-1):ON EY+1 GOTO 170
180 Y=20:POKE &HFEE4,X,Y:MEM$(&HFEE0,1)=MKI$(EX):MEM$(&HFEE1,1)=MKI$(EY)
190 POKE &HFEE6,10,10
200 POKE &HFEEE,0
210 '///// MAKE SCREEN /////
220 COLOR 1:LOCATE0,0:PRINTSTRING$(80,&H83):CGEN1:LOCATE0,23:PRINTSTRING$(80,&H8
3);:CGEN
230 COLOR 2:LOCATE1,1:PRINTSTRING$(22,HEXCHR$("861F1D"));
240 COLOR 2:LOCATE2,1:PRINTSTRING$(22,HEXCHR$("861F1D"));
250 CSIZE2:CGEN1:LOCATE40,24:PRINT#0,"[PLAYER 2:       0]";:CGEN
260 LOCATE74,10:PRINTL$;
270 COLOR 4:LOCATE78,1:PRINTSTRING$(22,HEXCHR$("861F1D"));
280 COLOR 4:LOCATE77,1:PRINTSTRING$(22,HEXCHR$("861F1D"));
290 CSIZE2:CGEN1:LOCATE0,24:PRINT#0,"[PLAYER 1:       0]";:CGEN
300 LOCATE5,10:PRINTL$;
310 GOSUB 590
320 A=STICK(0):B=STICK(1)
330 IF A+B=0 THEN 320 ELSE A=0
340 '///// MAIN ROUTINE /////
350 A=USR(A)
360 ON A GOTO 500,520,380
370 END
380 '///// GAME OVER /////
390 SOUND 7,56:PLAY "O2C7":LINE(18,8)-(60,14)," ",BF
400 IF PEEK(&HFEE4)<40 THEN SC2#=SC2#+500 ELSE SC1#=SC1#+500
410 CGEN1:CSIZE2:COLOR7:LOCATE18,10:PRINT#0,"THE WINNER IS ";
420 IF SC1#=SC2# THEN COLOR 5:PRINT#0,"NOBODY.":GOTO450
430 IF SC1#>SC2# THEN COLOR 4:PRINT#0,"PLAYER 1":GOTO450
440 COLOR 2:PRINT#0,"PLAYER 2"
450 COLOR 7:LOCATE24,12:PRINT#0,"TRY AGAIN ? (Y/N)"
460 A$=INKEY$(0):IF A$="Y" THEN WIDTH 80:INIT:CLS4:GOTO 20
470 IF A$="N" THEN END
480 GOTO 460
490 '///// SCORE /////
500 IF PEEK(&HFEE4)<40 THEN SC2#=SC2#+50 ELSE SC1#=SC1#+50
510 GOSUB550:GOTO350
520 IF PEEK(&HFEE0)=1 THEN SC1#=SC1#+1 ELSE SC2#=SC2#+1
530 GOSUB550:GOTO350
540 '///// PRINT SCORE /////
550 CSIZE2:CGEN1:COLOR2:LOCATE60,24:PRINT#0,USING"#######0";SC2#;
560 COLOR4:LOCATE20,24:PRINT#0,USING"#######0";SC1#;
570 RETURN
580 '///// STAGE /////
590 C=1:CGEN1:CSIZE
600 FOR I=5 TO 18 STEP 2
610 COLOR C:LOCATE22,I:PRINTSTRING$(36,&H80);
620 C=C+1:NEXT:CGEN
630 RETURN
640 '///// PCG /////
650 FORI=&H20 TO &H5F:A$=LEFT$(CGPAT$(I),8):B$=""
660 FORJ=1TO8:A=ASC(MID$(A$,J,1)):B=A OR A*2:IF B>255 THEN B=B-128:IF B>255 THEN
 B=B-128
670 B$=B$+CHR$(B):NEXT
680 DEFCHR$(I)=B$+B$+B$:NEXT
690 DEFCHR$(32)=STRING$(24,0)
700 DEFCHR$(48)=HEXCHR$(STRING$(3,"7CC6C6C6C6C67C00"))
710 DEFCHR$(77)=HEXCHR$(STRING$(3,"C6EEFED6C6C6C600"))
720 DEFCHR$(87)=HEXCHR$(STRING$(3,"C6C6C6D6FEEEC600"))
730 DEFCHR$(128)=HEXCHR$("FEFEFEFEFEFEFE00FEFEFEFEFEFEFE00FEFEFEFEFEFEFE00")
740 DEFCHR$(131)=HEXCHR$("FFFFFFFF00000000FFFFFFFF00000000FFFFFFFF00000000")
750 RETURN
```

リスト2　マシン語ルーチン

```
FA00 22 E8 FE 6E 26 00 29 11 : D6
FA08 10 FA 19 7E 23 66 6F E9 : 82
FA10 16 FA 8C FB 8C FB 3A EC : 44
FA18 FE B7 20 2C 3A E4 FE FE : 1B
FA20 05 30 03 AF 18 25 21 ED : 32
FA28 FE BE 38 03 AF 18 1C 3A : 14
FA30 E6 FE 87 3C 4F 3A E5 FE : 13
FA38 91 30 04 3E 38 18 0C 28 : 87
FA40 04 3E 32 18 06 AF 18 03 : 5C
FA48 CD 42 FD 21 E6 FE CD C2 : A0
FA50 FC 28 2B F5 08 CD 18 FD : 2E
FA58 01 05 00 09 44 4D 08 CD : 75
FA60 E2 FC D1 3A E6 FE 82 32 : 81
FA68 E6 FE 21 E6 FE CD 18 FD : CB
FA70 3E 04 32 17 FD 01 05 00 : 8E
FA78 09 44 4D CD FA FC 3A EC : 83
--------------------------------
SUM: 9D 9E 54 7A 70 63 DC DB CA4C

FA80 FE FE 02 CA B8 FA 3A E4 : 98
FA88 FE FE 4B 38 03 AF 18 2B : 74
FA90 3E 4A 21 E4 FE 96 21 ED : 2F
FA98 FE BE 38 03 AF 18 1C 3A : 14
FAA0 E7 FE 87 3C 4F 3A E5 FE : 14
FAA8 91 30 04 3E 38 18 0C 28 : 87
FAB0 04 3E 32 18 06 AF 18 03 : 5C
FAB8 CD 29 FD 21 E7 FE CD C2 : 88
FAC0 FC 28 2B F5 08 CD 18 FD : 2E
FAC8 01 4A 00 09 44 4D 08 CD : BA
FAD0 E2 FC D1 3A E7 FE 82 32 : 82
FAD8 E7 FE 21 E7 FE CD 18 FD : CD
FAE0 3E 02 32 17 FD 01 4A 00 : D1
FAE8 09 44 4D CD FA FC 3A E4 : 7B
FAF0 FE B7 CA 46 FC 3A E4 FE : DD
FAF8 FE 4F D2 46 FC ED 4B E2 : 7B
--------------------------------
SUM: 8A 51 98 2B FC 5F D2 DE C87E

FB00 FE 21 42 FE CD 97 FC 3A : F9
FB08 E5 FE 21 E1 FE 86 32 E5 : 80
FB10 FE 3A E4 FE 21 E0 FE 86 : 9F
FB18 32 E4 FE 2A E4 FE 22 3E : 80
FB20 FE 3E 20 32 9C FD CD 93 : 87
FB28 FD 3E 40 84 67 22 E2 FE : 68
FB30 44 4D 21 46 FE CD 97 FC : 56
FB38 AF 32 9C FD CD 93 FD 3E : 15
FB40 30 84 47 4D ED 78 32 40 : 1F
FB48 FE C5 F5 3A E4 FE 21 E0 : D5
FB50 FE 86 32 3E FE 3A E5 FE : 0F
FB58 21 E1 FE 86 32 3F FE AF : A4
FB60 32 9C FD CD 93 FD 3E 30 : 96
```

```
FB68 84 47 4D ED 78 FE 87 20 : 22
FB70 05 CD A0 FB 18 05 FE 83 : 0B
FB78 CC B9 FC F1 C1 FE 86 CA : 81
-----------------------------------
SUM: D5 51 B4 F1 83 67 10 18 251A

FB80 A6 FB FE 80 CA DC FB FE : BE
FB88 87 CC A0 FB 21 4A FE CB : 22
FB90 0E DC 4B FC ED 5B EA FE : 61
FB98 1B 7A B3 20 FB C3 16 FA : 36
FBA0 CD B0 FC C3 0B FE 2A E0 : 4F
FBA8 FE E5 CD B0 FC 3E 20 ED : A7
FBB0 79 CB A0 ED 78 E6 07 ED : 23
FBB8 79 CD 23 FC E1 FE 86 20 : EA
FBC0 0B 7C ED 44 67 7D ED 44 : CD
FBC8 6F 22 E0 FE CD E6 FD 3E : 5D
FBD0 01 C3 D4 FB 2A E8 FE 77 : 1A
FBD8 23 36 00 C9 21 EE FE 34 : 63
FBE0 3E 20 ED 79 CB A0 ED 78 : 94
FBE8 E6 07 F5 2A E0 FE E5 FE : CD
FBF0 02 20 05 CD B0 FC 18 0C : C4
FBF8 FE 04 20 05 CD B0 FC 18 : B8
-----------------------------------
SUM: D5 2C D0 6E DA E7 9C 62 E1E1

FC00 03 CD B9 FC AF ED 79 CD : 67
FC08 23 FC E1 FE 80 20 0B 7C : 25
FC10 ED 44 67 7D ED 44 6F 22 : D7
FC18 E0 FE F1 CD B7 FD 3E 02 : 90
FC20 C3 D4 FB 3A E4 FE 21 E0 : AF
FC28 FE 86 32 3E FE 3A E5 FE : 0F
FC30 21 E1 FE 86 32 3F FE AF : A4
FC38 32 9C FD CD 93 FD 3E 30 : 96
FC40 84 47 4D ED 78 C9 3E 03 : 87
FC48 C3 D4 FB 01 51 30 CD 60 : 41
FC50 FC 01 9E 30 CD 60 FC 01 : F5
FC58 52 30 CD 60 FC 01 9D 30 : 79
FC60 11 50 00 CB A0 ED 78 F5 : 26
FC68 CB E0 ED 78 F5 60 69 06 : D4
FC70 15 C5 19 44 4D ED 78 08 : F1
FC78 CB A0 ED 78 B7 ED 52 44 : 0A
-----------------------------------
SUM: 58 C3 C0 8C A5 43 C2 05 1B21

FC80 4D ED 79 CB E0 08 ED 79 : CC
FC88 19 C1 10 E5 F1 44 4D ED : 3E
FC90 79 CB A0 F1 ED 79 C9 04 : 08
FC98 ED A3 78 C6 08 47 04 ED : 0E
FCA0 A3 78 C6 08 47 04 ED A3 : C4
FCA8 78 C6 08 47 04 ED A3 C9 : EA
FCB0 3A E0 FE ED 44 32 E0 FE : 59
FCB8 C9 3A E1 FE ED 44 32 E1 : 26
FCC0 FE C9 FE 38 20 0E 7E FE : A7
FCC8 01 20 01 37 38 04 3E FF : D2
FCD0 18 01 AF C9 FE 32 20 08 : E9
FCD8 7E FE 14 30 03 3E 01 C9 : CB
FCE0 AF C9 FE FF 20 09 60 69 : 67
FCE8 11 50 00 19 19 44 4D 3E : 62
FCF0 20 ED 79 CB A0 3E 07 ED : 23
FCF8 79 C9 11 50 00 3E 03 08 : EC
-----------------------------------
SUM: D8 2B 98 3C 74 BE 3D 0C 032D

FD00 3E 87 ED 79 CB A0 3A 17 : E7
FD08 FD ED 79 CB E0 60 69 19 : F0
FD10 44 4D 08 3D 20 E9 C9 00 : A8
FD18 6E 26 00 29 29 29 29 54 : 8C
FD20 5D 29 29 19 3E 30 84 67 : 21
FD28 C9 C5 06 1C 3E 0E ED 79 : 62
FD30 05 ED 78 C1 0F 38 03 3E : B3
FD38 38 C9 0F 38 03 3E 32 C9 : 84
FD40 AF C9 C5 D5 E5 3A 91 FD : BF
FD48 32 92 FD FB 21 90 FD 16 : 80
FD50 E6 CD 6A FD CD 7C FD F3 : 53
FD58 CD 73 FD 72 23 CD 73 FD : 0F
FD60 72 23 FB 3A 91 FD E1 D1 : 0A
FD68 C1 C9 CD 7C FD 01 00 19 : EA
FD70 ED 51 C9 CD 86 FD 01 00 : 58
FD78 19 ED 50 C9 01 01 1A ED : 28
-----------------------------------
SUM: 1D 50 2E 63 8D D5 35 45 EE7D

FD80 78 E6 40 20 FA C9 01 01 : 83
FD88 1A ED 78 E6 20 20 FA C9 : 68
FD90 00 00 00 3A 3F FE 4F E6 : AC
FD98 01 28 02 3E 20 F5 CB 39 : 82
FDA0 26 00 69 29 29 29 29 44 : 77
FDA8 4D 29 29 09 F1 84 67 3A : BE
FDB0 3E FE 5F 16 00 19 C9 87 : 1A
FDB8 87 87 87 16 00 5F CD 36 : 0D
FDC0 FE 11 01 01 CD 36 FE 11 : 23
FDC8 38 07 CD 36 FE 11 10 08 : 69
FDD0 CD 36 FE 11 00 0B CD 36 : 20
FDD8 FE 11 0A 0C CD 36 FE 11 : 37
FDE0 00 0D CD 36 FE C9 11 04 : EC
FDE8 07 CD 36 FE 11 14 06 CD : 00
FDF0 36 FE 11 10 08 CD 36 FE : 5E
FDF8 11 00 0B CD 36 FE 11 14 : 42
-----------------------------------
SUM: 1A E0 27 41 78 31 72 67 EF54

FE00 0C CD 36 FE 11 00 0D CD : F8
FE08 36 FE C9 11 00 00 CD 36 : 11
FE10 FE 11 0A 01 CD 36 FE 11 : 2C
FE18 38 07 CD 36 FE 11 10 08 : 69
FE20 CD 36 FE 11 00 0B CD 36 : 20
FE28 FE 11 0A 0C CD 36 FE 11 : 37
FE30 00 0D CD 36 FE C9 06 1C : F9
FE38 ED 51 05 ED 59 C9 00 00 : 52
FE40 00 00 00 00 00 00 3C FF : 3B
FE48 FF 3C 55 00 3D CA 54 84 : 6F
FE50 09 C3 4C 84 C9 D8 3A 2E : A5
FE58 8A FE 10 C2 65 84 3A B9 : 36
FE60 D3 CD 85 88 C9 3A B9 D3 : 3C
FE68 CD 8F 88 C9 01 00 00 CD : 7B
FE70 8A 85 CD B2 86 CD 57 86 : BE
FE78 CD B2 86 C5 CD 39 5A C1 : EB
-----------------------------------
SUM: B9 18 C1 94 88 80 27 D0 0F9F

FE80 CA 6F 84 C5 CD 17 5B C1 : 82
FE88 3A 5D AC FE 1C CA C5 84 : 70
FE90 FE 1D CA D8 84 FE 1E CA : 27
FE98 EB 84 FE 1F CA FD 84 FE : D5
FEA0 20 CA 23 85 FE 0D CA 10 : 77
FEA8 85 FE 0C CA 73 85 FE 0B : 5A
FEB0 CA 73 85 FE 1B CA 81 85 : AB
FEB8 FE 13 CA 1B 85 C5 CD 3B : 48
FEC0 18 C1 C3 6F 84 3E 80 32 : 7F
FEC8 BD D3 04 21 2E 8A 78 BE : A3
FED0 C2 6F 84 06 00 C3 6F 84 : 71
FED8 78 FE 00 C2 E2 84 21 2E : ED
FEE0 01 01 00 40 28 19 0A 0A : 97
FEE8 00 00 00 10 00 00 00 F5 : 05
FEF0 84 21 2E 8A 4E 0D AF 32 : 99
FEF8 BD D3 C3 6F 84 3E 40 32 : F6
-----------------------------------
SUM: AB B1 B2 C3 D6 70 59 ED B7F7
```

リスト3　マシン語ルーチンのソースプログラム(参考)

```
                  1 ;-------------------------------
                  2        ; MAD BALL CLUB DX
                  3 ;-------------------------------
                  4        OFFSET  8000H-0FA00H
                  5        ORG     0FA00H
0005:             6 RX1:   EQU     5         ;ラケットのX
004A:             7 RX2:   EQU     74
                  8        ;A=USR(n)         ;OUT : n=FUNCTION NUMBER
                  9
FA00 22 E8 FE    10        LD      (HENSU),HL      ;#2
FA03 6E          11        LD      L,(HL)
FA04 26 00       12        LD      H,0
FA06 29          13        ADD     HL,HL   ;*2
FA07 11 10 FA    14        LD      DE,JUMP_TABLE
FA0A 19          15        ADD     HL,DE
FA0B 7E          16        LD      A,(HL)
FA0C 23          17        INC     HL
FA0D 66          18        LD      H,(HL)
FA0E 6F          19        LD      L,A
FA0F E9          20        JP      (HL)
                 21
FA10:            22 JUMP_TABLE:
FA10 16 FA 8C    23        DW      LOOP,]],]]
FA13 FB 8C FB
                 24
FA16:            25 LOOP:
FA16 3A EC FE    26        LD      A,(PLAYER)
FA19 B7          27        OR      A
FA1A 20 2C       28        JR      NZ,HUMAN1
                 29
FA1C 3A E4 FE    30        LD      A,(BX)
FA1F FE 05 30    31        IF      A<RX1   THEN XOR A JR HOHO
FA22 03 AF 18
FA25 25
FA26 21 ED FE    32        LD      HL,LEVEL
FA29 BE          33        CP      (HL)
FA2A 38 03 AF    34        IF      NC      THEN XOR A JR HOHO
FA2D 18 1C
FA2F 3A E6 FE    35        LD      A,(RY1)
FA32 87          36        ADD     A,A
FA33 3C          37        INC     A
FA34 4F          38        LD      C,A
FA35 3A E5 FE    39        LD      A,(BY)
FA38 91          40        SUB     C
FA39 30 04 3E    41        IF      C       THEN LD A,"8" JR HOHO
FA3C 38 18 0C
FA3F 28 04 3E    42        IF      NZ      THEN LD A,"2" JR HOHO
FA42 32 18 06
FA45 AF          43        XOR     A
FA46 18 03       44        JR      HOHO
                 45
FA48:            46 HUMAN1:
FA48 CD 3F FD    47        CALL    KEYIN
FA4B:            48 HOHO:
FA4B 21 E6 FE    49        LD      HL,RY1
FA4E CD C2 FC    50        CALL    OP_RY   ;A=VETCOR
FA51 28 2B       51        JR      Z,SKIP_RY1
FA53 F5          52        PUSH    AF
FA54 08          53        EX      AF,AF'
FA55 CD 15 FD    54        CALL    CALC_AD
FA58 01 05 00    55        LD      BC,RX1
FA5B 09          56        ADD     HL,BC
FA5C 44 4D       57        LD      BC,HL
FA5E 08          58        EX      AF,AF'
                 59
FA5F CD DF FC    60        CALL    KESU
FA62 D1          61        POP     DE
FA63 3A E6 FE    62        LD      A,(RY1)
FA66 82          63        ADD     A,D
FA67 32 E6 FE    64        LD      (RY1),A
FA6A 21 E6 FE    65        LD      HL,RY1
FA6D CD 15 FD    66        CALL    CALC_AD
FA70 3E 04       67        LD      A,4
FA72 32 14 FD    68        LD      (RCOL),A
FA75 01 05 00    69        LD      BC,RX1
FA78 09          70        ADD     HL,BC
FA79 44 4D       71        LD      BC,HL
FA7B CD F7 FC    72        CALL    KAKU
FA7E:            73 SKIP_RY1:
                 74        ;PLAYER CHECK
FA7E 3A EC FE    75        LD      A,(PLAYER)
FA81 FE 02 CA    76        IF      A=2     JP      HUMAN
FA84 B8 FA
                 77
FA86 3A E4 FE    78        LD      A,(BX)
FA89 FE 4B 38    79        IF      A>=RX2+1 THEN XOR A JR GOGO
FA8C 03 AF 18
FA8F 2B
FA90 3E 4A       80        LD      A,RX2
FA92 21 E4 FE    81        LD      HL,BX
FA95 96          82        SUB     (HL)
FA96 21 ED FE    83        LD      HL,LEVEL
FA99 BE          84        CP      (HL)
FA9A 38 03 AF    85        IF      NC THEN XOR A JR GOGO
FA9D 18 1C
FA9F 3A E7 FE    86        LD      A,(RY2)
FAA2 87          87        ADD     A,A
FAA3 3C          88        INC     A
FAA4 4F          89        LD      C,A
FAA5 3A E5 FE    90        LD      A,(BY)
FAA8 91          91        SUB     C
FAA9 30 04 3E    92        IF      C       THEN LD A,"8" JR GOGO
FAAC 38 18 0C
FAAF 28 04 3E    93        IF      NZ      THEN LD A,"2" JR GOGO
FAB2 32 18 06
FAB5 AF          94        XOR     A
FAB6 18 03       95        JR      GOGO
FAB8:            96 HUMAN:
FAB8 CD 26 FD    97        CALL    JOYSTICK
FABB:            98 GOGO:
FABB 21 E7 FE    99        LD      HL,RY2
FABE CD C2 FC   100        CALL    OP_RY   ;A=VETCOR
FAC1 28 2B      101        JR      Z,SKIP_RY2
FAC3 F5         102        PUSH    AF
FAC4 08         103        EX      AF,AF'
FAC5 CD 15 FD   104        CALL    CALC_AD
FAC8 01 4A 00   105        LD      BC,RX2
FACB 09         106        ADD     HL,BC
FACC 44 4D      107        LD      BC,HL
FACE 08         108        EX      AF,AF'
                109
FACF CD DF FC   110        CALL    KESU
FAD2 D1         111        POP     DE
FAD3 3A E7 FE   112        LD      A,(RY2)
FAD6 82         113        ADD     A,D
FAD7 32 E7 FE   114        LD      (RY2),A
FADA 21 E7 FE   115        LD      HL,RY2
FADD CD 15 FD   116        CALL    CALC_AD
FAE0 3E 02      117        LD      A,2
FAE2 32 14 FD   118        LD      (RCOL),A
FAE5 01 4A 00   119        LD      BC,RX2
FAE8 09         120        ADD     HL,BC
FAE9 44 4D      121        LD      BC,HL
                122
FAEB CD F7 FC   123        CALL    KAKU
FAEE:           124 SKIP_RY2:
                125
                126
                127        ;       B A L L ′ S   M O V E M E N T
                128
FAEE 3A E4 FE   129        LD      A,(BX)
FAF1 B7         130        OR      A
FAF2 CA 46 FC   131        JP      Z,OVER
FAF5 3A E4 FE   132        LD      A,(BX)
FAF8 FE 4F D2   133        IF      A>=79   JP OVER
FAFB 46 FC
```

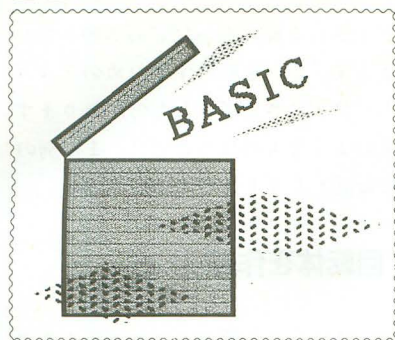

▶2月号の表紙の色使いの渋さには感動しました。いつもなら本屋で「Oh!Xはどこだー」という感じで探すのに，今回は本屋へ行くなり「この渋い本はなんだ！」，「オー！Xやー」と，この本のタイトルどおりの感銘を受けました。んー，渋い。阿部　貴秀 (19) 石川県

```
                   134
FAFD ED 4B E2      135          LD      BC,(LAST_ADR)
FB00 FE
FB01 21 3F FE      136          LD      HL,B_KESU
FB04 CD 97 FC      137          CALL    PUTOUT
                   138
FB07 3A E5 FE      139          LD      A,(BY)
FB0A 21 E1 FE      140          LD      HL,EY_MARK
FB0D 86            141          ADD     A,(HL)  ;A=NEW BY
FB0E 32 E5 FE      142          LD      (BY),A  ;STORE IT
                   143
FB11:              144 SKIP_EY:
FB11 3A E4 FE      145          LD      A,(BX)
FB14 21 E0 FE      146          LD      HL,EX_MARK
FB17 86            147          ADD     A,(HL)
FB18 32 E4 FE      148          LD      (BX),A
                   149
FB1B:              150 SKIP_EX:
FB1B 2A E4 FE      151          LD      HL,(BX)
FB1E 22 3B FE      152          LD      (BX'),HL
                   153
FB21 3E 20         154          LD      A,32
FB23 32 99 FD      155          LD      (PATCH+1),A
FB26 CD 90 FD      156          CALL    BALL_ADR
FB29 3E 40         157          LD      A,40H
FB2B 84            158          ADD     A,H
FB2C 67            159          LD      H,A
FB2D 22 E2 FE      160          LD      (LAST_ADR),HL
FB30 44 4D         161          LD      BC,HL
FB32 21 43 FE      162          LD      HL,B_PAT
FB35 CD 97 FC      163          CALL    PUTOUT
                   164
FB38 AF            165          XOR     A
FB39 32 99 FD      166          LD      (PATCH+1),A
FB3C CD 90 FD      167          CALL    BALL_ADR
FB3F 3E 30         168          LD      A,30H
FB41 84            169          ADD     A,H
FB42 47            170          LD      B,A
FB43 4D            171          LD      C,L
FB44 ED 78         172          IN      A,(C)   ;NOW_POSITION
FB46 32 3D FE      173          LD      (NOW_POS),A
FB49 C5            174          PUSH    BC
FB4A F5            175          PUSH    AF
                   176
FB4B 3A E4 FE      177          LD      A,(BX)
FB4E 21 E0 FE      178          LD      HL,EX_MARK
FB51 86            179          ADD     A,(HL)
FB52 32 3B FE      180          LD      (BX'),A
                   181
FB55 3A E5 FE      182          LD      A,(BY)
FB58 21 E1 FE      183          LD      HL,EY_MARK
FB5B 86            184          ADD     A,(HL)
FB5C 32 3C FE      185          LD      (BY'),A
                   186
FB5F AF            187          XOR     A
FB60 32 99 FD      188          LD      (PATCH+1),A
FB63 CD 90 FD      189          CALL    BALL_ADR
FB66 3E 30         190          LD      A,30H
FB68 84            191          ADD     A,H
FB69 47            192          LD      B,A
FB6A 4D            193          LD      C,L
FB6B ED 78         194          IN      A,(C)   ;NEXT POSITION
                   195
FB6D FE 87 20      196          IF      A=87H   THEN CALL   HIT_R      JR [[
FB70 05 CD A0
FB73 FB 18 05
FB76 FE 83 CC      197          IF      A=83H   CALL    OPSITY
FB79 B9 FC
FB7B:F1            198 [[:      POP     AF
FB7C C1            199          POP     BC
FB7D FE 86 CA      200          IF      A=86H   JP      HIT_BRR
FB80 A6 FB
FB82 FE 80 CA      201          IF      A=80H   JP      HIT_BLK
FB85 DC FB
FB87 FE 87 CC      202          IF      A=87H   CALL    HIT_R
FB8A A0 FB
FB8C:              203 ]]:
FB8C 21 47 FE      204          LD      HL,RWK
FB8F CB 0E         205          RRC     (HL)
FB91 DC 4B FC      206          CALL    C,ROLL
                   207
FB94 ED 5B EA      208          DO      DE,(WAIT_TIME) ( )
FB97 FE 1B 7A
FB9A B3 20 FB
                   209
FB9D C3 16 FA      210          JP      LOOP
FBA0:              211 HIT_R:
FBA0 CD B0 FC      212          CALL    OPSITX
FBA3 C3 08 FE      213          JP      SE3
                   214
FBA6:              215 HIT_BRR:
FBA6 2A E0 FE      216          LD      HL,(EX_MARK)
FBA9 E5            217          PUSH    HL
FBAA CD B0 FC      218          CALL    OPSITX
FBAD 3E 20         219          LD      A," "
FBAF ED 79         220          OUT     (C),A
FBB1 CB A0         221          RES     4,B
FBB3 ED 78         222          IN      A,(C)
FBB5 E6 07         223          AND     7
FBB7 ED 79         224          OUT     (C),A
FBB9 CD 23 FC      225          CALL    READ_NEXT
FBBC E1            226          POP     HL
FBBD FE 86 20      227          IF      A=86H   THEN
FBC0 0B
FBC1 7C            228          LD      A,H
FBC2 ED 44         229          NEG
FBC4 67            230          LD      H,A
FBC5 7D            231          LD      A,L
FBC6 ED 44         232          NEG
FBC8 6F            233          LD      L,A
FBC9 22 E0 FE      234          LD      (EX_MARK),HL
                   235          FI
                   236
FBCC CD E3 FD      237          CALL    SE2
FBCF 3E 01         238          LD      A,1
FBD1 C3 D4 FB      239          JP      RET     ;NEXT GOTO ]]
                   240
FBD4:              241 RET:
FBD4 2A E8 FE      242          LD      HL,(HENSU)
FBD7 77            243          LD      (HL),A
FBD8 23            244          INC     HL
FBD9 36 00         245          LD      (HL),0
FBDB C9            246          RET
                   247
FBDC:              248 HIT_BLK:
FBDC 21 EE FE      249          LD      HL,BREAK
FBDF 34            250          INC     (HL)
FBE0 3E 20         251          LD      A," "
FBE2 ED 79         252          OUT     (C),A
FBE4 CB A0         253          RES     4,B
FBE6 ED 78         254          IN      A,(C)
FBE8 E6 07         255          AND     7
FBEA F5            256          PUSH    AF
                   257
FBEB 2A E0 FE      258          LD      HL,(EX_MARK)
FBEE E5            259          PUSH    HL
                   260
FBEF FE 02 20      261          IF      A=2  THEN
FBF2 05
FBF3 CD B0 FC      262          CALL    OPSITX
FBF6 18 0C         263          ELSE
FBF8 FE 04 20      264          IF      A=4  THEN
FBFB 05
FBFC CD B0 FC      265          CALL    OPSITX
FBFF 18 03         266          ELSE
FC01 CD B9 FC      267          CALL    OPSITY
                   268          FI
                   269          FI
FC04 AF            270          XOR     A
FC05 ED 79         271          OUT     (C),A                ;NEXT GOTO ]]
                   272
FC07 CD 23 FC      273          CALL    READ_NEXT
FC0A E1            274          POP     HL
FC0B FE 80 20      275          IF      A=80H   THEN
FC0E 0B
FC0F 7C            276          LD      A,H
FC10 ED 44         277          NEG
FC12 67            278          LD      H,A
FC13 7D            279          LD      A,L
FC14 ED 44         280          NEG
FC16 6F            281          LD      L,A
FC17 22 E0 FE      282          LD      (EX_MARK),HL
                   283          FI
                   284
FC1A F1            285          POP     AF
FC1B CD B4 FD      286          CALL    SE1
FC1E 3E 02         287          LD      A,2
FC20 C3 D4 FB      288          JP      RET
                   289
FC23:              290 READ_NEXT:
FC23 3A E4 FE      291          LD      A,(BX)
FC26 21 E0 FE      292          LD      HL,EX_MARK
FC29 86            293          ADD     A,(HL)
FC2A 32 3B FE      294          LD      (BX'),A
FC2D 3A E5 FE      295          LD      A,(BY)
FC30 21 E1 FE      296          LD      HL,EY_MARK
FC33 86            297          ADD     A,(HL)
FC34 32 3C FE      298          LD      (BY'),A
FC37 AF            299          XOR     A
FC38 32 99 FD      300          LD      (PATCH+1),A
FC3B CD 90 FD      301          CALL    BALL_ADR
FC3E 3E 30         302          LD      A,30H
FC40 84            303          ADD     A,H
FC41 47            304          LD      B,A
FC42 4D            305          LD      C,L
FC43 ED 78         306          IN      A,(C)
FC45 C9            307          RET
FC46:              308 OVER:
FC46 3E 03         309          LD      A,3
FC48 C3 D4 FB      310          JP      RET
                   311
FC4B:              312 ROLL:
FC4B 01 51 30      313          LD      BC,3000H+80+1
FC4E CD 60 FC      314          CALL    ROLL_SUB
FC51 01 9E 30      315          LD      BC,3000H+80+78
FC54 CD 60 FC      316          CALL    ROLL_SUB
FC57 01 52 30      317          LD      BC,3000H+80+2
FC5A CD 60 FC      318          CALL    ROLL_SUB
FC5D 01 9D 30      319          LD      BC,3000H+80+77
                   320
FC60:              321 ROLL_SUB:
FC60 11 50 00      322          LD      DE,80
FC63 CB A0         323          RES     4,B
FC65 ED 78         324          IN      A,(C)
FC67 F5            325          PUSH    AF
FC68 CB E0         326          SET     4,B
FC6A ED 78         327          IN      A,(C)
FC6C F5            328          PUSH    AF
FC6D 60 69         329          LD      HL,BC
FC6F 06 15         330          DO      B,21 {
FC71 C5            331          PUSH    BC
FC72 19            332          ADD     HL,DE
FC73 44 4D         333          LD      BC,HL
FC75 ED 78         334          IN      A,(C)
FC77 08            335          EX      AF,AF'
FC78 CB A0         336          RES     4,B
FC7A ED 78         337          IN      A,(C)
                   338
FC7C B7 ED 52      339          SUB     HL,DE
FC7F 44 4D         340          LD      BC,HL
FC81 ED 79         341          OUT     (C),A
FC83 CB E0         342          SET     4,B
FC85 08            343          EX      AF,AF'
FC86 ED 79         344          OUT     (C),A
FC88 19            345          ADD     HL,DE
FC89 C1            346          POP     BC
FC8A 10 E5         347          }
FC8C F1            348          POP     AF
FC8D 44 4D         349          LD      BC,HL
FC8F ED 79         350          OUT     (C),A
FC91 CB A0         351          RES     4,B
FC93 F1            352          POP     AF
FC94 ED 79         353          OUT     (C),A
FC96 C9            354          RET
                   355
FC97:              356 PUTOUT:          ;ボールを描く
FC97 04 ED A3      357          INC     B       OUTI
FC9A 78 C6 08      358          LD      A,B     ADD A,8 LD B,A
FC9D 47
FC9E 04 ED A3      359          INC     B       OUTI
FCA1 78 C6 08      360          LD      A,B     ADD A,8 LD B,A
FCA4 47
FCA5 04 ED A3      361          INC     B       OUTI
FCA8 78 C6 08      362          LD      A,B     ADD A,8 LD B,A
FCAB 47
FCAC 04 ED A3      363          INC     B       OUTI
FCAF C9            364          RET
                   365
FCB0:              366 OPSITX:
FCB0 3A E0 FE      367          LD      A,(EX_MARK)
FCB3 ED 44         368          NEG
FCB5 32 E0 FE      369          LD      (EX_MARK),A
FCB8 C9            370          RET
FCB9:              371 OPSITY:
FCB9 3A E1 FE      372          LD      A,(EY_MARK)
FCBC ED 44         373          NEG
FCBE 32 E1 FE      374          LD      (EY_MARK),A
FCC1 C9            375          RET
                   376
FCC2:              377 OP_RY:
                   378          ;< (HL)=RY
                   379          ;> A=EY
                   380
FCC2 FE 38 20      381          IF A="8" THEN
FCC5 0B
FCC6 7E            382                  LD      A,(HL)
FCC7 FE 02 38      383                  IF A>1 THEN
FCCA 04
FCCB 3E FF         384                  LD      A,-1
FCCD 18 01         385                  ELSE
FCCF AF            386                  XOR     A
                   387                  FI
FCD0 C9            388          RET
                   389          FI
FCD1 FE 32 20      390          IF A="2" THEN
FCD4 08
FCD5 7E            391                  LD      A,(HL)
FCD6 FE 14 30      392                  IF A<20 THEN
FCD9 03
FCDA 3E 01         393                  LD      A,1
FCDC C9            394                  RET
                   395                  FI
```

▶現在，スペハリの入力に備えて肩を休めております。　　西山　新志（17）福岡県

```
FCDD AF          396          FI
FCDE C9          397          XOR     A
FCDF:            398          RET
                 399 KESU:
                 400          ;< A=EY, BC=T_RAM
                 401          ;x HL DE
FCDF FE FF 20    402          IF      A=-1    THEN
FCE2 09
FCE3 60 69       403                  LD      HL,BC
FCE5 11 50 00    404                  LD      DE,80
FCE8 19          405                  ADD     HL,DE
FCE9 19          406                  ADD     HL,DE
FCEA 44 4D       407                  LD      BC,HL   ;BC=BC+2
                 408          FI
FCEC 3E 20       409          LD      A," "
FCEE ED 79       410          OUT     (C),A
FCF0 CB A0       411          RES     4,B
FCF2 3E 07       412          LD      A,7
FCF4 ED 79       413          OUT     (C),A   ;COLOR  7
FCF6 C9          414          RET
                 415
FCF7:            416 KAKU:
                 417          ;< BC=T_RAM
                 418          ;WORK : (RCOL)=COLOR
                 419          ;x ALL
FCF7 11 50 00    420          LD      DE,80
FCFA 3E 03       421          LD      A,3
                 422
FCFC:            423 RKTL:
FCFC 08          424          EX      AF,AF'
FCFD 3E 87       425          LD      A,87H
FCFF ED 79       426          OUT     (C),A
FD01 CB A0       427          RES     4,B
FD03 3A 14 FD    428          LD      A,(RCOL)
FD06 ED 79       429          OUT     (C),A
FD08 CB E0       430          SET     4,B
                 431
FD0A 60 69       432          LD      HL,BC
FD0C 19          433          ADD     HL,DE
FD0D 44 4D       434          LD      BC,HL
FD0F 08          435          EX      AF,AF'
FD10 3D          436          DEC     A
FD11 20 E9       437          JR      NZ,RKTL
FD13 C9          438          RET
FD14:00          439 RCOL:    DS      1
                 440
                 441
FD15:            442 CALC_AD:
                 443          ;< (HL)=RY
                 444          ;> HL=TRAM ADDR
                 445          ;x A,DE,HL
FD15 6E          446          LD      L,(HL)
FD16:            447 CALC_AD':
FD16 26 00       448          LD      H,0
FD18 29          449          ADD     HL,HL   ;*2
FD19 29          450          ADD     HL,HL   ;*4
FD1A 29          451          ADD     HL,HL   ;*8
FD1B 29          452          ADD     HL,HL   ;*16
FD1C 54 5D       453          LD      DE,HL
FD1E 29          454          ADD     HL,HL   ;*32
FD1F 29          455          ADD     HL,HL   ;*64
FD20 19          456          ADD     HL,DE   ;*80
FD21 3E 30       457          LD      A,30H
FD23 84          458          ADD     A,H
FD24 67          459          LD      H,A
FD25 C9          460          RET
                 461
FD26:            462 JOYSTICK:
                 463          ; > A=STICK DATA
                 464
FD26 C5          465          PUSH    BC
FD27 06 1C       466          LD      B,1CH
FD29 3E 0E       467          LD      A,0EH
FD2B ED 79       468          OUT     (C),A
FD2D 05          469          DEC     B
FD2E ED 78       470          IN      A,(C)
FD30 C1          471          POP     BC
FD31 0F          472          RRCA
FD32 38 03 3E    473          IF NC THEN LD A,"8" RET
FD35 38 C9
FD37 0F          474          RRCA
FD38 38 03 3E    475          IF NC THEN LD A,"2" RET
FD3B 32 C9
FD3D AF          476          XOR     A
FD3E C9          477          RET
                 478
FD3F:            479 KEYIN:
                 480   ;<  NONE
                 481   ;>  A=ASCII CODE  (0 MEANS NO KEY)
                 482   ;x  A
                 483   ;-  BC DE HL
                 484
FD3F C5          485          PUSH    BC
FD40 D5          486          PUSH    DE
FD41 E5          487          PUSH    HL
                 488
FD42 3A 8E FD    489          LD      A,(KASCII)
FD45 32 8F FD    490          LD      (LASTKY),A
FD48 FB          491          EI
FD49 21 8D FD    492          LD      HL,KEYCON
FD4C 16 E6       493          LD      D,0E6H          ;D=&HE6 (COM)
FD4E CD 67 FD    494          CALL    SEND1
FD51 CD 79 FD    495          CALL    CANW
FD54 F3          496          DI
                 497
FD55 CD 70 FD    498          CALL    GET1
FD58 72          499          LD      (HL),D
FD59 23          500          INC     HL
FD5A CD 70 FD    501          CALL    GET1
FD5D 72          502          LD      (HL),D
FD5E 23          503          INC     HL
FD5F FB          504          EI
                 505
                 506   ;HL=LASTKY ノ ADDRESS ニ ナッテイル.
                 507
FD60 3A 8E FD    508          LD      A,(KASCII)
FD63 E1          509          POP     HL
FD64 D1          510          POP     DE
FD65 C1          511          POP     BC
FD66 C9          512          RET
                 513
FD67:            514 SEND1
FD67 CD 79 FD    515          CALL    CANW
FD6A 01 00 19    516          LD      BC,01900H
FD6D ED 51       517          OUT     (C),D
FD6F C9          518          RET
                 519
FD70:            520 GET1
FD70 CD 83 FD    521          CALL    CANR
FD73 01 00 19    522          LD      BC,01900H
FD76 ED 50       523          IN      D,(C)
FD78 C9          524          RET
                 525
FD79:            526 CANW
FD79 01 01 1A    527          LD      BC,01A01H
FD7C:            528 CANWL
FD7C ED 78       529          IN      A,(C)
FD7E E6 40       530          AND     040H
FD80 20 FA       531          JR      NZ,CANWL
FD82 C9          532          RET
                 533
FD83:            534 CANR
FD83 01 01 1A    535          LD      BC,01A01H
FD86:            536 CANRL
FD86 ED 78       537          IN      A,(C)
FD88 E6 20       538          AND     020H
FD8A 20 FA       539          JR      NZ,CANRL
FD8C C9          540          RET
                 541
FD8D:00          542 KEYCON:          DS 1
FD8E:00          543 KASCII:          DS 1
FD8F:00          544 LASTKY:          DS 1
                 545
FD90:            546 BALL_ADR:
                 547          ; WORK : BY' & BX'
                 548
                 549
                 550
FD90 3A 3C FE    551          LD      A,(BY')
FD93 4F          552          LD      C,A
FD94 E6 01       553          AND     00000001B
FD96 28 02       554          JR      Z,ZRO
FD98:            555 PATCH:
FD98 3E 20       556          LD      A,8*4
FD9A:            557 ZRO:
FD9A F5          558          PUSH    AF      ;#1
FD9B CB 39       559          SRL     C
FD9D 26 00       560          LD      H,0
FD9F 69          561          LD      L,C
FDA0 29          562          ADD     HL,HL   ;*2
FDA1 29          563          ADD     HL,HL   ;*4
FDA2 29          564          ADD     HL,HL   ;*8
FDA3 29          565          ADD     HL,HL   ;*16
FDA4 44 4D       566          LD      BC,HL
FDA6 29          567          ADD     HL,HL   ;*32
FDA7 29          568          ADD     HL,HL   ;*64
FDA8 09          569          ADD     HL,BC   ;*80
FDA9 F1          570          POP     AF      ;#1
FDAA 84          571          ADD     A,H
FDAB 67          572          LD      H,A
FDAC 3A 3B FE    573          LD      A,(BX')
FDAF 5F          574          LD      E,A
FDB0 16 00       575          LD      D,0
FDB2 19          576          ADD     HL,DE
FDB3 C9          577          RET
FDB4:            578 SE1:
                 579          ;< A=NUMBER
FDB4 87          580          ADD     A,A
FDB5 87          581          ADD     A,A
FDB6 87          582          ADD     A,A
FDB7 87          583          ADD     A,A
FDB8 16 00       584          LD      D,0
FDBA 5F          585          LD      E,A
FDBB CD 33 FE    586          WPSG
FDBE 11 01 01    587          LD      DE,0101H        WPSG
FDC1 CD 33 FE
FDC4 11 38 07    588          LD      DE,7*256+56     WPSG
FDC7 CD 33 FE
FDCA 11 10 08    589          LD      DE,8*256+16     WPSG
FDCD CD 33 FE
FDD0 11 00 0B    590          LD      DE,11*256       WPSG
FDD3 CD 33 FE
FDD6 11 0A 0C    591          LD      DE,12*256+10    WPSG
FDD9 CD 33 FE
FDDC 11 00 0D    592          LD      DE,13*256       WPSG
FDDF CD 33 FE
FDE2 C9          593          RET
FDE3:            594 SE2:
FDE3 11 04 07    595          LD      DE,0704H        WPSG
FDE6 CD 33 FE
FDE9 11 14 06    596          LD      DE,6*256+20     WPSG
FDEC CD 33 FE
FDEF 11 10 08    597          LD      DE,8*256+16     WPSG
FDF2 CD 33 FE
FDF5 11 00 0B    598          LD      DE,11*256       WPSG
FDF8 CD 33 FE
FDFB 11 14 0C    599          LD      DE,12*256+20    WPSG
FDFE CD 33 FE
FE01 11 00 0D    600          LD      DE,13*256       WPSG
FE04 CD 33 FE
FE07 C9          601          RET
FE08:            602 SE3:
FE08 11 00 00    603          LD      DE,0    WPSG
FE0B CD 33 FE
FE0E 11 0A 01    604          LD      DE,010AH        WPSG
FE11 CD 33 FE
FE14 11 38 07    605          LD      DE,7*256+56     WPSG
FE17 CD 33 FE
FE1A 11 10 08    606          LD      DE,8*256+16     WPSG
FE1D CD 33 FE
FE20 11 00 0B    607          LD      DE,11*256       WPSG
FE23 CD 33 FE
FE26 11 0A 0C    608          LD      DE,12*256+10    WPSG
FE29 CD 33 FE
FE2C 11 00 0D    609          LD      DE,13*256       WPSG
FE2F CD 33 FE
FE32 C9          610          RET
FE33:            611 WPSG:
FE33 06 1C       612          LD      B,1CH
FE35 ED 51       613          OUT     (C),D
FE37 05          614          DEC     B
FE38 ED 59       615          OUT     (C),E
FE3A C9          616          RET
FE3B:00          617 BX':     DS      1
FE3C:00          618 BY':     DS      1
FE3D:00          619 NOW_POS:         DS      1
FE3E:00          620 NEXT_POS:        DS      1
FE3F:00 00 00    621 B_KESU:          DB      0,0,0,0
FE42 00
FE43:3C          622 B_PAT:           DB      00111100B
FE44 FF          623                  DB      11111111B
FE45 FF          624                  DB      11111111B
FE46 3C          625                  DB      00111100B
FE47:55          626 RWK:     DB      01010101B
                 627
                 628          ORG     0FEE0H
FEE0:01          629 EX_MARK:         DB      1
FEE1:01          630 EY_MARK:         DB      1
                 631
FEE2:00 40       632 LAST_ADR:        DW      4000H
                 633
FEE4:28          634 BX:      DB      40      ;ボールのX
FEE5:19          635 BY:      DB      25      ;ボールのY
                 636
FEE6:0A          637 RY1:     DB      10      ;ラケットのY
FEE7:0A          638 RY2:     DB      10
                 639
FEE8:00 00       640 HENSU: DW      0       ;SAVE USR ADDR
FEEA:00 10       641 WAIT_TIME:       DW      1000H
FEEC:00          642 PLAYER:          DS      1
FEED:00          643 LEVEL:           DS      1
FEEE:00          644 BREAK:           DS      1

ハッシュ TBL: 008A
シンボルTBL: 02BF
ストラクトTBL: 0058

オブジェクト: FA00 [8000] - FEEE [84EE]
```

入門3Dグラフィック

計算で作る立体データと隠面処理

グラフィックには初心者なら初心者なりの，上級者にはそれにあわせた楽しみ方があります。3Dグラフィックもまずは BASIC で初級編からいってみましょう。

Misawa Kazuhiko
三沢 和彦

私が愛機X1マニアタイプを手に入れたときグラフィックの美しさに感激したのをいまでも覚えています。特に数学に出てくる関数のグラフなどをBASICを使って計算，表示させたときの美しさには目を見張るものがあります。グラフィックツールなどで絵を描くのもいいのですが，コンピュータで計算させて作った画像には独特の魅力があるように思われます。今回はBASICでコンピュータグラフィックの真骨頂ともいうべき3次元グラフィックに挑戦してみましょう。

正多面体の3D表示

グラフィックではまさに「百聞は一見にしかず」ですから，まずはリスト1を入力して実行してみてください。これは3次元図形のうちでももっとも規則的な正多面体を表示するものです。X1用のBASIC (CZ-8FB01)で記述されていますが，他機種への移植もそんなに難しくはないでしょう。turbo BASIC ならなんの変更も必要ありません。

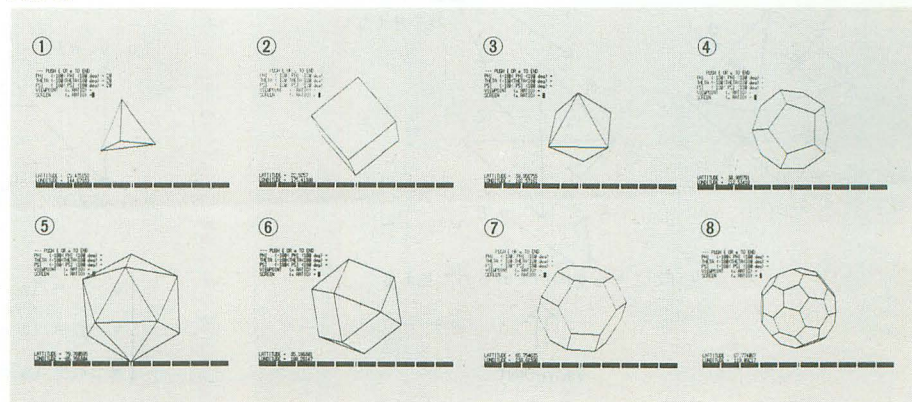
リスト1は各種の正多面体を3D表示するプログラムです。正多面体というのは，各面が同一の正多角形のみで構成され，しかもすべての頂点で集まる辺の数が同数の凸型立体のことをいいます。

正多面体は構造が単純かつ美しく，手作業で頂点を求めるのもそれほど苦にはなりません。ここでは，4個の正3角形で作られる正4面体，6個の正方形で作られる正6面体，8個の正3角形を組み合わせた正8面体，12個の正6角形を組み合わせた正12面体，20個の正3角形を組み合わせた正20面体，そして正多面体ではありませんが，正3角形と正方形を組み合わせた14面体，正6角形と正方形を組み合わせた14面体，そして正5角形，正6角形を組み合わせたサッカーボール立体（？）の8種類が登録されています。

このリスト1の自信作はなんといってもサッカーボールです。正5角形と正6角形を組み合わせた非常に魅惑的な形をしています。かなり球形に近くなっていますが，見る角度によってはずいぶん角張っても見えます。データ入力には苦労しましたが，皆さんはできるだけ間違えないように注意して入力してください。

立体を回してみる

起動すると，メニューが現れ，先ほどの8種類の立体のどれを表示するかを聞いてきます。1から8のキーでメニューから立体を選択するとデータを読み込み，基本位置の図形を3D表示します。

試しに1番の正4面体を選ぶと，データの読み込みを始め，しばらくすると3角形1個を表示します。これは，どうしたことでしょう。実は，ある特定の方向から見ると正4面体も3角形1個にしか見えないということです。このプログラムでは実際の視点からの映像と同じように，物体の陰に隠れた面は表示しないようになっているのです。

画面左上には回転角を指定するように入力待ちの状態でカーソルが点滅しているはずです。ここから各軸回りに回転角を指定することで，さまざまな角度から眺めることができます。−180〜180までの数値で指定してください。

ここで用いた3つの回転角PHI, THETA, PSI (ϕ, θ, ψ) をオイラー角といいますが，どの角がどこに相当するのかについては図1を参照してください。このオイラー角を理論的に理解するには専門的な知識が必要なのであまり深く考えないほうがよいかもしれません。

3つの角度を指定すると今度は，

VIEWPOINT (X RATIO)=

と表示されます。ここで視点から立体までの距離の比を設定します。2倍に遠ざかるときは2，半分の距離に近づくときは0.5を入力してください。

さらに，

SCREEN (X RATIO)=

というぐあいに入力待ちの状態になります。これはスクリーン上の拡大/縮少率です。さっきのVIEWPOINTで2倍に遠ざかると，画面の物体が半分の大きさになってしまうので，相似拡大する場合などに使います。

実行例

この2つの指定のどこがどういうふうに違うのかと疑問に思われる方もいるかもしれません。VIEWPOINTでは視点の移動を伴うので大きさとともに全体的なパースも変化するのに対し，SCREENの指定では形はそのまま大きさだけが変化します。

適当に角度を変えてやると，より立体図形らしく表示されます。いろいろと試してみてください。なお，これらの指定は現在の画面に対しての指定となりますので，操作を続けると回転がどんどん積み重ねられていきます。最初のメニューに戻るには角度入力の部分でEキーを押します。

◇ 隠面処理

このプログラムで使用している隠面処理の判断などはコンピュータの得意な計算を駆使しています。これこそコンピュータグラフィックの醍醐味といえましょう。

ただ，隠面処理といっても，今回行ったものは比較的，処理の簡単な凸多面体の隠面処理に限定していますから，それほど複雑なことにはなりません。いきなり複雑なことをやっても消化不良になりそうですから，まずは，基本部分をしっかり押さえておきましょう。

表示するのが凸多面体の場合は，各面に垂直な矢印を立て，その矢印が視点の方向に向いているかどうかで判断を行います。(図2参照)。矢印がこっちを向いている場合は必ずその面が見えているのです。さて，この矢印を決めるには図3のような公式を使います（実はベクトルの外積を求めているのですが，高校生以下の方や文系大学生にはお馴染みがない概念だと思います。とりあえずは，こういう公式があるんだと割り切って読んでください)。

今回は視点がZ軸上にあるので矢印がZ軸と同じ方向を向いているか（平行でなくてもよい）を判断するには，$(x_2-x_1)(y_3-y_1)-(x_3-x_1)(y_2-y_1)$ の符号を調べます。これが正のときにはZ軸と同じ方向，負のときには逆の方向を向いているわけです。リスト1，2では1020行がそういった判断を行っています。わからなければ，このあたりは読み飛ばしておいてもかまいません(かなり面倒です)。

凸多面体の場合はすべての面についてこの処理を繰り返せばことは足りるのですが，へこみのある凹多面体の場合は，たとえその面が視点方向を向いていても，ほかの面の陰によって隠されてしまう場合があるので，表示される面のあいだでの処理が加わってきます。それを解決するためのアルゴリズムには実にさまざまなものがありますが，そのようなものについては，また別の機会にお話ししましょう。

◇ 回転体を作る

リスト2はもっと遊べるプログラムです。これまでは手作業で求めた立体の形状データを与えてそれを計算/表示していただけですが，ここでは立体の頂点データを求める部分も自動化しています。といっても，まったく規則性のない形状は計算によって求めることができませんから，一定の法則性を持ったもののみを扱うことにします。ここでは回転体の頂点データを生成することにしました。

図1　オイラー角

図2　隠面処理

図3　表面を決める矢印

面の表を向いている矢印の決め方
(ベクトルの外積)

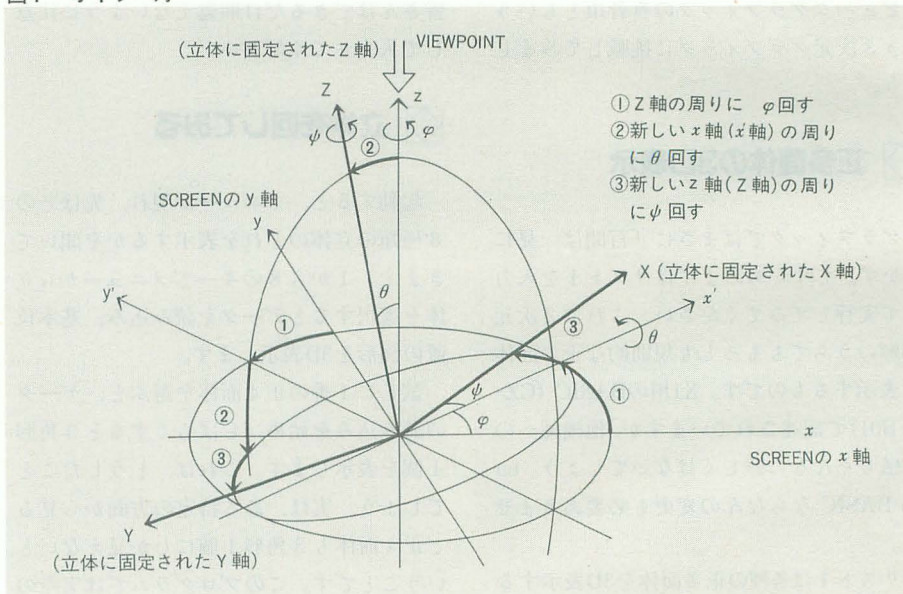
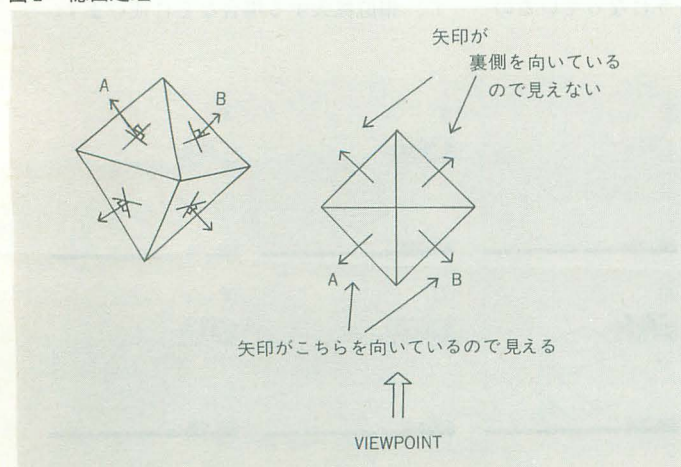
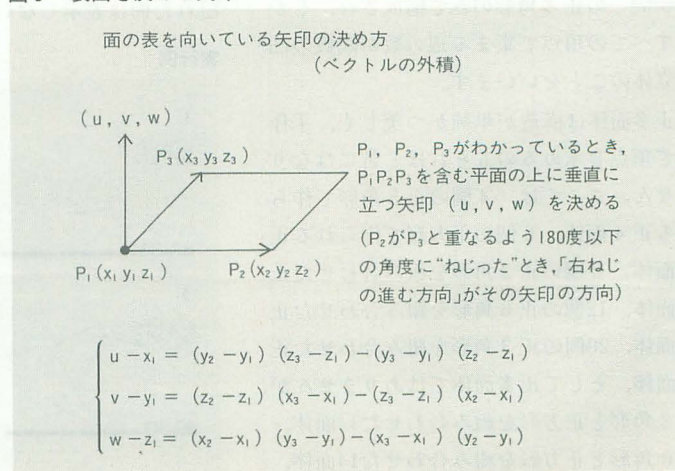
(u, v, w)

$P_3(x_3\ y_3\ z_3)$

$P_1(x_1\ y_1\ z_1)$　$P_2(x_2\ y_2\ z_2)$

P_1, P_2, P_3がわかっているとき，$P_1P_2P_3$を含む平面の上に垂直に立つ矢印（u, v, w）を決める

(P_2がP_3と重なるよう180度以下の角度に“ねじった”とき,「右ねじの進む方向」がその矢印の方向)

$$\begin{cases} u-x_1=(y_2-y_1)(z_3-z_1)-(y_3-y_1)(z_2-z_1) \\ v-y_1=(z_2-z_1)(x_3-x_1)-(z_3-z_1)(x_2-x_1) \\ w-z_1=(x_2-x_1)(y_3-y_1)-(x_3-x_1)(y_2-y_1) \end{cases}$$

回転体とはある平面図形を軸のまわりに回転させたとき，その平面が通過する空間によってできる図形のことですが，これなら断面図を適当に描いて与えるだけで，立体のデータを算定できます。

リスト２を実行すると，最初に赤い２次元の座標軸と原点に置かれた白い丸印が表示されます。縦がＺ軸，横がＹ軸になっていて，回転体はＺ軸のまわりに作られるようになっています。

テンキーの２，４，６，８を使って，回転体の断面の各項点を決定します。適当なところまで白丸を動かして５キーで決定してください。なお，処理の都合上，凸多面体しか作成できないようになっていますので注意してください。４つか５つくらいの点を打ち込んだら，Ｚ軸上に点を打って断面の入力を終了します。

DEVISION＝

の指示に対して，適当な数値を入力してください。これは回転体とはいっても，完全な円周を持つものは扱えないので正多角形で代用しているためです。円周を16くらいに分割すると綺麗に表示されます。あまりに大きな数値を入れるとエラーが発生しますので気をつけましょう。

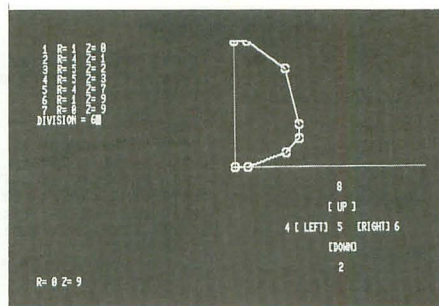
初期状態では回転体を真上から見ていることになるので，円グラフのようなものが表示されるはずです。

“おもちゃ”としてのパソコン

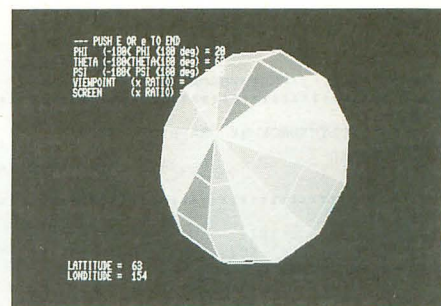
今回は3Dグラフィックといって初級のものを扱いました。隠面処理もより複雑な立体に対処するためにはもっと高度な処理法が要求されてきます。しかし，今回の２つのプログラムだけでもある程度3Dグラフィックの楽しさがわかっていただけたかと思います。

こういったぐあいに，パソコンというものは自分の入力したプログラムに従って，思いどおりの処理をさせられる点に“おもちゃ”としての面白さがあるのでしょう。たとえ，それがごく簡単な処理であっても同じことです。もともと，“おもちゃ”だと思えば，特に高度なことをする必要はありません。

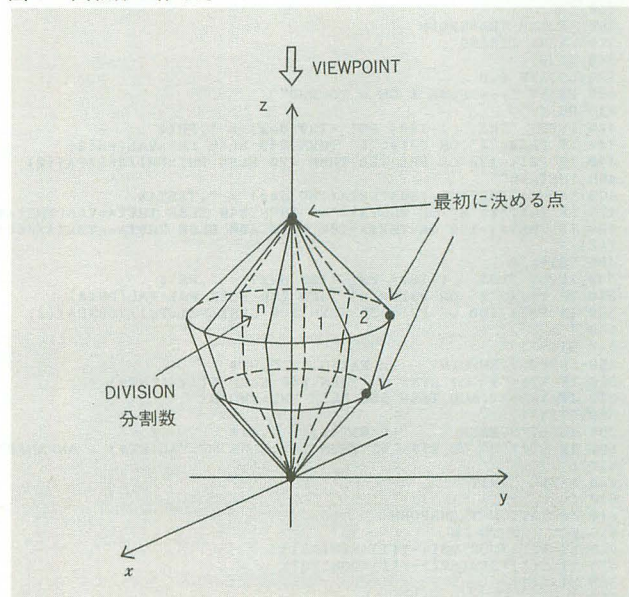
最初のうちは，

PRINT “A”

のような単純なプログラムでも，それが自分の意思によって表示されるというところに喜びを感じるものでしょう。

そして，パソコンの素晴しいところは，プログラミングする自分の技術進歩や知識の拡大に伴って，どこまでも高度な処理ができる可能性を秘めていることでしょう。最初のうちは稚拙なプログラムでも，ユーザーの成長により，見違えるようなプログラムもできるようになるのです。そして，それを操作しているのは自分だという充実感もあります。

BASICという言語はプログラミング言語としては，いろいろと問題点が指摘されていますが，やはり入門者にとっては扱いやすいということの利点のほうが大きいでしょう。特にプログラムしていて，どうしてもBASICでは困るという重大局面に出くわさない限りにおいては，BASICでも十分なのです。初心者の皆さんもBASICを使って自らマシンを操るという実感を楽しんでみてください。

このようにエディットする

回転体の実行結果

図４　回転体の作り方

データの応用

さて，このようにして作った3Dデータも，ただ回しているだけでは面白くありません。データを生かす方法としてどのようなものがあるかを考えてみましょう。

ここで行ったことをよく見てみると，回転体データの生成などは，非常にお手軽な3Dモデリングでもあるわけです。そもそも３次元グラフィックでもっとも面倒なのが，データの生成すなわちモデリングです。

3Dツールを使っていてデータに不自由したことはありませんか？　たとえば，そう，turbo R AYTRACER，とかMAGICとか。これらは非常に優秀な 3D グラフィックパッケージなのですが，データを入力していくのはなかなか手間がかかるので使いこなせてない人もいるのではないでしょうか。

ということで，いかにこれをモデリングツールとして応用していくか。ここからは数学の得意な人への自由課題としておきます。

たとえばturbo RAYTRACERなら，平面のプリミティブを使って，立体を作っていくのがよいでしょう。ここで扱えるのは凸型立体のみですが，逆にいえば出力されるのはすべて凸型立体ですから，各面で単純に論理演算を行えば，望みの立体が手に入るわけです。

turbo RAYTRACERで必要なパラメータは，

$ax+by+cz=d$

の平面の方程式の各係数となっていますから，頂点の座標からこれを導かねばなりません（頂点データは配列に入っていますね)。さらに物体の中と外を決めるため法線ベクトルの方向も特定しなければなりません。一見，難しそうですが，これは隠面処理の際に行われる裏表を決定するアルゴリズムを応用できるでしょう。

MAGICなら，もっと単純に頂点データを拾い出せます。平面のデータから結合関係を導いていくことはそれほど困難ではないでしょう。それでは，皆さんがんばってみてください。

リスト1 正多面体

```
10 '  SAVE "1:POLYHEDRON1.BAS
20 '**************************************************************
30 '
40 '   POLYHEDRON 3D GRAPHICS
50 '
60 '                    1989. 1.14   K.MISAWA
70 '
80 '**************************************************************
90 '
100 WIDTH 80 : INIT
110 WINDOW (0,0)-(639,199),(-5,3)-(5,-3)      :'WINDOW
120 I=80 : J=10 : K=50
130 DIM PX(I),PY(I),PZ(I)                     :'INITIAL POINTS
140 DIM X(I),Y(I),Z(I)                :'POINTS
150 DIM SX(I),SY(I),SZ(I)             :'POINTS ON SCREEN
160 DIM PL(K,J)                       :'SURFACE DATA
170 '
180 VWD=10                            :'DISTANCE VIEWPOINT TO OBJECT
190 SCD=3                             :'DISTANCE VIEWPOINT TO SCREEN
200 WD=.1                             :'WINDOW WIDTH
210 AN=WD/SCD                         :'VIEW ANGLE
220 X(0)=0 : Y(0)=0 : Z(0)=1          :'Z AXIS UNIT VECTOR
230 '
240 GOSUB "DATA"
250 '
260 GOSUB "INIT"
270 GOSUB "NVEC"
280 'GOSUB "PALET"
290 '
300 GOSUB "TRANS"
310 GOSUB "NVEC"
320 'GOSUB "PALET"
330 '
340 GOTO 300
350 '
360 'EULAR TRANSFORM
370 LABEL "TRANS"
380 KEY0,""
390 LOCATE 0,0
400 PRINT "--- PUSH E OR e TO END"
410 PHI$=""
420 INPUT "PHI   (-180< PHI <180 deg) = ",PHI$
430 IF PHI$="E" OR PHI$="e" THEN 240 ELSE PHI=VAL(PHI$)
440 IF PHI<-180 OR PHI>180 THEN 420 ELSE PHI=PHI/360*PAI(2)
450 THETA$=""
460 INPUT "THETA (-180<THETA<180 deg) = ",THETA$
470 IF THETA$="E" OR THETA$="e" THEN 240 ELSE THETA=VAL(THETA$)
480 IF THETA<-180 OR THETA>180 THEN 460 ELSE THETA=-THETA/360*PA
I(2)
490 PSI$=""
500 INPUT "PSI   (-180< PSI <180 deg) = ",PSI$
510 IF PSI$="E" OR PSI$="e" THEN 240 ELSE PSI=VAL(PSI$)
520 IF PSI<-180 OR PSI>180 THEN 500 ELSE PSI=PSI/360*PAI(2)
530 '
540 RT$="1"
550 INPUT"VIEWPOINT    (x RATIO) = ",RT$
560 IF RT$="E" OR RT$="e" THEN 240 ELSE RT=VAL(RT$)
570 IF VWD*RT<SCD THEN 550 ELSE VWD=VWD*RT
580 RT$="1"
590 INPUT"SCREEN       (x RATIO) = ",RT$
600 IF RT$="E" OR RT$="e" THEN 240 ELSE RT=VAL(RT$) : AN=AN/RT
610 '
620 LABEL "INIT"
630 '
640 'ROTATION TRANSFORM
650 FOR I=0 TO PN
660 X1=X(I)*COS(PSI)-Y(I)*SIN(PSI)
670 Y1=X(I)*SIN(PSI)+Y(I)*COS(PSI)
680 Z1=Z(I)
690 '
700 X2=X1
710 Y2=Y1*COS(THETA)-Z1*SIN(THETA)
720 Z2=Y1*SIN(THETA)+Z1*COS(THETA)
730 '
740 X(I)=X2*COS(PHI)-Y2*SIN(PHI)
750 Y(I)=X2*SIN(PHI)+Y2*COS(PHI)
760 Z(I)=Z2
770 '
780 '2D TRANSFORM
790 LABEL "VIEW"
800  DZ=VWD-Z(I)
810  SX(I)=X(I)/DZ/AN
820  SY(I)=Y(I)/DZ/AN
830  SZ(I)=Z(I)/DZ/AN
840 NEXT
850 RETURN
860 '
870 'NORMAL VECTOR
880 LABEL "NVEC"
890 CLS 4
900 '
910 'Z AXIS DIRECTION
920 DIST=X(0)*X(0)+Y(0)*Y(0)
930 IF DIST=0 THEN LAT=90*SGN(Z(0)) ELSE LAT=ATN(Z(0)/SQR(DIST))
/PAI(2)*360
940 IF X(0)=0 THEN LON=90*SGN(Y(0)) ELSE IF X(0)>0 THEN LON=ATN(
Y(0)/X(0))/PAI(2)*360 ELSE LON=180+ATN(Y(0)/X(0))/PAI(2)*360
950 LOCATE0,22 : PRINT "LATTITUDE = ";LAT
960 LOCATE0,23 : PRINT "LONDITUDE = ";LON;
970 '
980 FOR L=1 TO PLN
990  PL=PL(L,1) : XA=SX(PL) : YA=SY(PL)
1000  PL=PL(L,2) : XB=SX(PL) : YB=SY(PL)
1010  PL=PL(L,3) : XC=SX(PL) : YC=SY(PL)
1020  NV=SGN((XB-XA)*(YC-YA)-(XC-XA)*(YB-YA))    :'VECTOR PRODUCT
1030  IF NV=1 THEN GOSUB "DRAW"                  :'CHECK VISIBLE
1040 NEXT L
1050 RETURN
1060 '
1070 'DRAW SURFACE
1080 LABEL "DRAW"
1090 PL1=PL(L,1) : PL2=PL(L,2) : PL3=PL(L,3)
1100 LINE(SX(PL1),SY(PL1))-(SX(PL2),SY(PL2))
1110 FOR I=3 TO PL(L,0)
1120  LINE -(SX(PL(L,I)),SY(PL(L,I)))
1130 NEXT
1140 LINE-(SX(PL1),SY(PL1))
1150 RETURN
1160 '
1170 LABEL "DATA"
1180 CLS 4
1190 PRINT "SELECT POLYHEDRON NUMBER"
1200 PRINT "  [1]  4 POLYHEDRON
1210 PRINT "  [2]  6 POLYHEDRON
1220 PRINT "  [3]  8 POLYHEDRON
1230 PRINT "  [4] 12 POLYHEDRON
1240 PRINT "  [5] 20 POLYHEDRON
1250 PRINT "  [6] 14 POLYHEDRON (1)
1260 PRINT "  [7] 14 POLYHEDRON (2)
1270 PRINT "  [8] SOCCER BALL
1280 LOCATE 0,9: PRINT CHR$(5);: INPUT" POLYHEDRON NUMBER = ";TN
1290 IF TN<1 OR TN>8 THEN GOTO 1280
1300 '
1310 CLS
1320 IF TN=1 THEN RESTORE "4POLY"
1330 IF TN=2 THEN RESTORE "6POLY"
1340 IF TN=3 THEN RESTORE "8POLY"
1350 IF TN=4 THEN RESTORE "12POLY"
1360 IF TN=5 THEN RESTORE "20POLY"
1370 IF TN=6 THEN RESTORE "14POLY1"
1380 IF TN=7 THEN RESTORE "14POLY2"
1390 IF TN=8 THEN RESTORE "SOCCER"
1400 '
1410 'DATA READ
1420 PRINT "READING DATA --- WAIT A MINUTE!"
1430 READ SCALE
1440 READ PN                          :'POINT TOTAL NUMBER
1450 FOR I=1 TO PN
1460  READ X,Y,Z                      :'POINTS DATA
1470  PX(I)=X/SCALE : PY(I)=Y/SCALE : PZ(I)=Z/SCALE
1480 X(I)=X/SCALE : Y(I)=Y/SCALE : Z(I)=Z/SCALE
1490 NEXT I
1500 READ PLN                         :'SURFACE TOTAL NUMBER
1510 FOR J=1 TO PLN
1520  READ PL(J,0)                    :'SURFACE POINTS
1530  FOR I=1 TO PL(J,0)
1540   READ PL(J,I)                   :'SURFACE DATA
1550  NEXT I
1560 NEXT J
1570 RETURN
1580 '
1590 'DATA
1600 LABEL "4POLY"
1610 DATA 1                                               :'SCALE
1620 DATA 4                                               :'PN
1630 DATA -.5,0,.289,.5,0,.289,0,0,-.577,0,.816,0         :'PX,PY,PZ
1640 DATA 4                                               :'PLN
1650 DATA 4, 1,2,4,1                                      :'PL
1660 DATA 4, 2,3,4,2
1670 DATA 4, 3,1,4,3
1680 DATA 4, 1,3,2,1
1690 '
1700 LABEL "6POLY"
1710 DATA 2
1720 DATA 8
1730 DATA -1,-1,-1,-1,-1,1,1,-1,1,1,-1,-1,-1,1,-1,-1,1,1,1,1,1,1
,1,-1
1740 DATA 6
1750 DATA 5, 1,2,6,5,1
1760 DATA 5, 3,4,8,7,3
1770 DATA 5, 5,6,7,8,5
1780 DATA 5, 1,4,3,2,1
1790 DATA 5, 2,3,7,6,2
1800 DATA 5, 4,1,5,8,4
1810 '
1820 LABEL "8POLY"
1830 DATA 1.4142
1840 DATA 6
1850 DATA -1,0,0,0,0,1,1,0,0,0,0,-1,0,1,0,0,-1,0
1860 DATA 8
1870 DATA 4, 1,2,5,1
1880 DATA 4, 2,3,5,2
1890 DATA 4, 3,4,5,3
1900 DATA 4, 4,1,5,4
1910 DATA 4, 1,6,2,1
1920 DATA 4, 2,6,3,2
1930 DATA 4, 3,6,4,3
1940 DATA 4, 4,6,1,4
1950 '
1960 LABEL "12POLY"
1970 DATA 2.472136
1980 DATA 20
1990 DATA -1,1,-1,-1,1,1,-1,-1,1,-1,-1,-1,1,1,-1,1,1,1,1,-1,1,1,
-1,-1
2000 DATA -1.618,.618,0,-1.618,-.618,0,1.618,.618,0,1.618,-.618,
0
2010 DATA 0,1.618,-.618,0,1.618,.618,0,-1.618,-.618,0,-1.618,.61
8
2020 DATA -.618,0,1.618,.618,0,1.618,-.618,0,-1.618,.618,0,-1.61
8
```

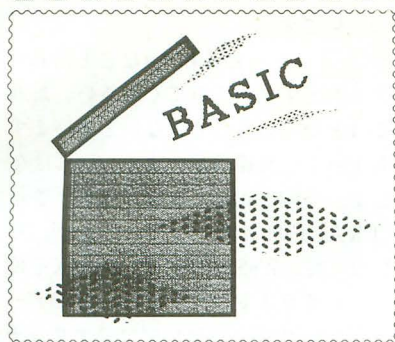
▶12月号の新作ソフト情報で,「シルバーゴーストを新しいタイプのRPGだ」と書いてありましたが,このソフトのタイプは,あの「ボコスカウォーズ」じゃないんですか?
岡野　博（17）埼玉県

```
2030 DATA 12
2040 DATA 6, 1,9,2,14,13,1
2050 DATA 6, 2,9,10,3,17,2
2060 DATA 6, 2,17,18,6,14,2
2070 DATA 6, 6,11,5,13,14,6
2080 DATA 6, 1,13,5,20,19,1
2090 DATA 6, 1,19,4,10,9,1
2100 DATA 6, 3,16,7,18,17,3
2110 DATA 6, 6,18,7,12,11,6
2120 DATA 6, 4,19,20,8,15,4
2130 DATA 6, 3,10,4,15,16,3
2140 DATA 6, 8,12,7,16,15,8
2150 DATA 6, 5,11,12,8,20,5
2160 '
2170 LABEL "20POLY"
2180 DATA 1
2190 DATA 12
2200 DATA -.688,.425,-.5,-.688,.425,.5,.263,.425,.809
2210 DATA .851,.425,0,.263,.425,-.809,-.851,-.425,0
2220 DATA -.263,-.425,.809,.688,-.425,.5,.688,-.425,-.5
2230 DATA -.263,-.425,-.809,0,.951,0,0,-.951,0
2240 DATA 20
2250 DATA 4, 1,2,11,1
2260 DATA 4, 2,3,11,2
2270 DATA 4, 3,4,11,3
2280 DATA 4, 4,5,11,4
2290 DATA 4, 5,1,11,5
2300 DATA 4, 2,1,6,2
2310 DATA 4, 2,6,7,2
2320 DATA 4, 2,7,3,2
2330 DATA 4, 3,7,8,3
2340 DATA 4, 3,8,4,3
2350 DATA 4, 4,8,9,4
2360 DATA 4, 4,9,5,4
2370 DATA 4, 5,9,10,5
2380 DATA 4, 5,10,1,5
2390 DATA 4, 1,10,6,1
2400 DATA 4, 7,6,12,7
2410 DATA 4, 8,7,12,8
2420 DATA 4, 9,8,12,9
2430 DATA 4, 10,9,12,10
2440 DATA 4, 6,10,12,6
2450 '
2460 LABEL "14POLY1"
2470 DATA 1.4142
2480 DATA 12
2490 DATA 1,0,-1,0,1,-1,-1,0,-1,0,-1,-1
2500 DATA 1,1,0,-1,1,0,-1,-1,0,1,-1,0
2510 DATA 1,0,1,0,1,1,-1,0,1,0,-1,1
2520 DATA 14
2530 DATA 5, 1,5,9,8,1
2540 DATA 5, 2,6,10,5,2
2550 DATA 5, 3,7,11,6,3
2560 DATA 5, 4,8,12,7,4
2570 DATA 5, 10,11,12,9,10
2580 DATA 5, 1,4,3,2,1
2590 DATA 4, 1,2,5,1
2600 DATA 4, 2,3,6,2
2610 DATA 4, 3,4,7,3
2620 DATA 4, 4,1,8,4
2630 DATA 4, 5,10,9,5
2640 DATA 4, 6,11,10,6
2650 DATA 4, 7,12,11,7
2660 DATA 4, 8,9,12,8
2670 '
2680 LABEL "14POLY2"
2690 DATA 0.7140
2700 DATA 24
2710 DATA .333,0,-.667,0,.333,-.667,-.333,0,-.667,0,-.333,-.667
2720 DATA .667,0,-.333,0,.667,-.333,-.667,0,-.333,0,-.667,-.333
2730 DATA .667, .333,0, .333, .667,0,-.333, .667,0,-.667, .333,0
2740 DATA -.667,-.333,0,-.333,-.667,0, .333,-.667,0, .667,-.333,
0
2750 DATA .667,0,.333,0,.667,.333,-.667,0,.333,0,-.667,.333
2760 DATA .333,0,.667,0,.333,.667,-.333,0,.667,0,-.333,.667
2770 DATA 14
2780 DATA 7, 1,2,6,10,9,5,1
2790 DATA 7, 2,3,7,12,11,6,2
2800 DATA 7, 3,4,8,14,13,7,3
2810 DATA 7, 4,1,5,16,15,8,4
2820 DATA 7, 9,10,18,22,21,17,9
2830 DATA 7, 11,12,19,23,22,18,11
2840 DATA 7, 13,14,20,24,23,19,13
2850 DATA 7, 15,16,17,21,24,20,15
2860 DATA 5, 1,4,3,2,1
2870 DATA 5, 5,9,17,16,5
2880 DATA 5, 6,11,18,10,6
2890 DATA 5, 7,13,19,12,7
2900 DATA 5, 8,15,20,14,8
2910 DATA 5, 21,22,23,24,21
2920 '
2930 LABEL"SOCCER"
2940 DATA 1
2950 DATA 60
2960 DATA -.459,.600,.333,-.688,.425,.167,-.742,.142,.333,-.546,
.142,.603,-.371,.425,.603
2970 DATA -.459,.600,-.333,-.371,.425,-.603,-.546,.142,-.603,-.7
42,.142,-.333,-.688,.425,-.167
2980 DATA  .175,.600,-.539,.459,.425,-.539,.405,.142,-.706,.0877
,.142,-.809,-.054,.425,-.706
2990 DATA  .567,.600,0,.655,.425,.270,.797,.142,.167,.797,.142,-
.167,.655,.425,-.270
3000 DATA  .175,.600,.539,-.054,.425,.706,.0877,.142,.809,.405,.
142,.706,.459,.425,.539
3010 DATA -.567,-.600,0,-.655,-.425,.270,-.797,-.142,.167,-.797,
-.142,-.167,-.655,-.425,-.270
3020 DATA -.175,-.600,-.539,-.459,-.425,-.539,-.405,-.142,-.706,
-.0877,-.142,-.809,.054,-.425,-.706
3030 DATA  .459,-.600,-.333,.371,-.425,-.603,.546,-.142,-.603,.7
42,-.142,-.333,.688,-.425,-.167
3040 DATA  .459,-.600,.333, .688,-.425,.167,.742,-.142,.333,.546
,-.142,.603,.371,-.425,.603
3050 DATA -.175,-.600,.539,.054,-.425,.706,-.0877,-.142,.809,-.4
05,-.142,.706,-.459,-.425,.539
3060 DATA -.229,.776,.167,.0877,.776,.270,.284,.776,0,.0877,.776
,-.270,-.229,.776,-.167
3070 DATA -.284,-.776,0,-.0877,-.776,-.270,.229,-.776,-.167,.229
,-.776,.167,-.0877,-.776,.270
3080 DATA 32
3090 DATA 6, 1,2,3,4,5,1
3100 DATA 6, 6,7,8,9,10,6
3110 DATA 6, 11,12,13,14,15,11
3120 DATA 6, 16,17,18,19,20,16
3130 DATA 6, 21,22,23,24,25,21
3140 DATA 6, 26,27,28,29,30,26
3150 DATA 6, 31,32,33,34,35,31
3160 DATA 6, 36,37,38,39,40,36
3170 DATA 6, 41,42,43,44,45,41
3180 DATA 6, 46,47,48,49,50,46
3190 DATA 6, 51,52,53,54,55,51
3200 DATA 6, 56,57,58,59,60,56
3210 DATA 7, 2,1,51,55,6,10,2
3220 DATA 7, 7,6,55,54,11,15,7
3230 DATA 7, 12,11,54,53,16,20,12
3240 DATA 7, 17,16,53,52,21,25,17
3250 DATA 7, 22,21,52,51,1,5,22
3260 DATA 7, 28,3,2,10,9,29,28
3270 DATA 7, 30,29,9,8,33,32,30
3280 DATA 7, 33,8,7,15,14,34,33
3290 DATA 7, 35,34,14,13,38,37,35
3300 DATA 7, 38,13,12,20,19,39,38
3310 DATA 7, 40,39,19,18,43,42,40
3320 DATA 7, 43,18,17,25,24,44,43
3330 DATA 7, 45,44,24,23,48,47,45
3340 DATA 7, 48,23,22,5,4,49,48
3350 DATA 7, 50,49,4,3,28,27,50
3360 DATA 7, 56,26,30,32,31,57,56
3370 DATA 7, 57,31,35,37,36,58,57
3380 DATA 7, 58,36,40,42,41,59,58
3390 DATA 7, 59,41,45,47,46,60,59
3400 DATA 7, 60,46,50,27,26,56,60
```

リスト2　回転体を作る

```
10 '  SAVE "1:POLYHEDRON2.BAS
20 '**********************************************************
30 '
40 '   POLYHEDRON 3D GRAPHICS
50 '
60 '                        1989 .1.14   K.MISAWA
70 '
80 '**********************************************************
90 '
100 WIDTH 80 : INIT
110 WINDOW (0,0)-(639,199),(-5,3)-(5,-3)      :'WINDOW
120 I=100 : J=5 : K=130
130 DIM PX(I),PY(I),PZ(I)                     :'INITIAL POINTS
140 DIM X(I),Y(I),Z(I)                        :'POINTS
150 DIM SX(I),SY(I),SZ(I)                     :'POINTS ON SCREEN
160 DIM PL(K,J)                               :'SURFACE DATA
170 '
180 VWD=10                         :'DISTANCE VIEWPOINT TO OBJECT
190 SCD=3                          :'DISTANCE VIEWPOINT TO SCREEN
200 WD=.1                                     :'WINDOW WIDTH
210 AN=WD/SCD                                 :'VIEW ANGLE
220 X(0)=0 : Y(0)=0 : Z(0)=1                  :'Z AXIS UNIT VECTOR
230 '
240 GOSUB "DATA"
250 '
260 GOSUB "INIT"
270 GOSUB "NVEC"
280 GOSUB "PALET"
290 '
300 GOSUB "TRANS"
310 GOSUB "NVEC"
320 GOSUB "PALET"
330 '
340 GOTO 300
350 '
360 'EULAR TRANSFORM
370 LABEL "TRANS"
380 KEY0,""
390 LOCATE 0,0
400 PRINT"--- PUSH E OR e TO END "
410 PHI$=""
420 INPUT "PHI   (-180< PHI <180 deg) = ",PHI$
430 IF PHI$="E" OR PHI$="e" THEN 240 ELSE PHI=VAL(PHI$)
440 IF PHI<-180 OR PHI>180 THEN 420 ELSE PHI=PHI/360*PAI(2)
```

▶毎回，GAME REVIEWのあのわけのわかんないゲーム要素の円グラフを楽しく見させてもらってます。
福田　弘一（18）大阪府

```
450 THETA$=""
460 INPUT "THETA (-180<THETA<180 deg) = ",THETA$
470 IF THETA$="E" OR THETA$="e" THEN 240 ELSE THETA=VAL(THETA$)
480 IF THETA<-180 OR THETA>180 THEN 460 ELSE THETA=-THETA/360*PA
I(2)
490 PSI$=""
500 INPUT "PSI   (-180< PSI <180 deg) = ",PSI$
510 IF PSI$="E" OR PSI$="e" THEN 240 ELSE PSI=VAL(PSI$)
520 IF PSI<-180 OR PSI>180 THEN 500 ELSE PSI=PSI/360*PAI(2)
530 '
540 RT$="1"
550 INPUT"VIEWPOINT   (x RATIO) = ",RT$
560 IF RT$="E" OR RT$="e" THEN 240 ELSE RT=VAL(RT$)
570 IF VWD*RT<SCD THEN 550 ELSE VWD=VWD*RT
580 RT$="1"
590 INPUT"SCREEN      (x RATIO) = ",RT$
600 IF RT$="E" OR RT$="e" THEN 240 ELSE RT=VAL(RT$) :AN=AN/RT
610 '
620 LABEL "INIT"
630 '
640 'ROTATION TRANSFORM
650 FOR I=0 TO PN
660 X1=X(I)*COS(PSI)-Y(I)*SIN(PSI)
670 Y1=X(I)*SIN(PSI)+Y(I)*COS(PSI)
680 Z1=Z(I)
690 '
700 X2=X1
710 Y2=Y1*COS(THETA)-Z1*SIN(THETA)
720 Z2=Y1*SIN(THETA)+Z1*COS(THETA)
730 '
740 X(I)=X2*COS(PHI)-Y2*SIN(PHI)
750 Y(I)=X2*SIN(PHI)+Y2*COS(PHI)
760 Z(I)=Z2
770 '
780 '2D TRANSFORM
790 LABEL "VIEW"
800  DZ=VWD-Z(I)
810  SX(I)=X(I)/DZ/AN
820  SY(I)=Y(I)/DZ/AN
830  SZ(I)=Z(I)/DZ/AN
840 NEXT
850 RETURN
860 '
870 'NORMAL VECTOR
880 LABEL "NVEC"
890 CLS 4
900 '
910 'Z AXIS DIRECTION
920 DIST=X(0)*X(0)+Y(0)*Y(0)
930 IF DIST=0 THEN LAT=90*SGN(Z(0)) ELSE LAT=ATN(Z(0)/SQR(DIST))
/PAI(2)*360
940 IF X(0)=0 THEN LON=90*SGN(Y(0)) ELSE IF X(0)>0 THEN LON=ATN(
Y(0)/X(0))/PAI(2)*360 ELSE LON=180+ATN(Y(0)/X(0))/PAI(2)*360
950 LOCATE0,22 : PRINT "LATTITUDE = ";LAT
960 LOCATE0,23 : PRINT "LONDITUDE = ";LON;
970 '
980 FOR L=1 TO PLN
990  PL=PL(L,1) : XA=SX(PL) : YA=SY(PL)
1000  PL=PL(L,2) : XB=SX(PL) : YB=SY(PL)
1010  PL=PL(L,3) : XC=SX(PL) : YC=SY(PL)
1020  NV=SGN((XB-XA)*(YC-YA)-(XC-XA)*(YB-YA))      :'VECTOR PRODUCT
1030  IF NV=1 THEN GOSUB "DRAW"                    :'CHECK VISIBLE
1040 NEXT L
1050 RETURN
1060 '
1070 'DRAW SURFACE
1080 LABEL "DRAW"
1090 PL1=PL(L,1) : PL2=PL(L,2) :PL3=PL(L,3)
1100 LINE(SX(PL1),SY(PL1))-(SX(PL2),SY(PL2))
1110 FOR I=3 TO PL(L,0)
1120  LINE -(SX(PL(L,I)),SY(PL(L,I)))
1130 NEXT
1140 LINE-(SX(PL1),SY(PL1))
1150 '
1160 XS1=SX(PL2)-SX(PL1) : XS2=SX(PL3)-SX(PL1)
1170 YS1=SY(PL2)-SY(PL1) : YS2=SY(PL3)-SY(PL1)
1180 SCP=XS1*XS2+YS1*YS2 : SCP=SCP*SCP
1190 NRM=(XS1*XS1+YS1*YS1)*(XS2*XS2+YS2*YS2)
1200 IF ABS(SCP-NRM)<.0001 THEN 1240
1210 PTX=(SX(PL1)+SX(PL2)+SX(PL3))/3
1220 PTY=(SY(PL1)+SY(PL2)+SY(PL3))/3
1230 PAINT(PTX,PTY),(L MOD 6)+1,7
1240 RETURN
1250 '
1260 'PALET
1270 LABEL "PALET
1280 PLT=0
1290 LOCATE 0,0 : PRINT "--- HIT ANY KEY TO STOP"
1300 KEY0,"" : Z$=""
1310 FOR CL=1 TO 6
1320  PALET CL,((CL+PLT) MOD 6)+1
1330 NEXT
1340 PLT=PLT+1
1350 Z$=INKEY$ : IF Z$="" THEN 1310
1360 RETURN
1370 '
1380 'INPUT DATA
1390 LABEL "DATA"
1400 CLS 4
1410 GOSUB "2DDATA"
1420 '
1430 PX(1)=0 : PY(1)=0 : PZ(1)=0
1440 FOR I=1 TO M-1
1450  FOR J=0 TO N-1
1460   PTN=N*(I-1)+J+2
1470   PX(PTN)=R(I)*COS(PAI(2)/N*J)/ZI(M) : X(PTN)=PX(PTN)
1480   PY(PTN)=R(I)*SIN(PAI(2)/N*J)/ZI(M) : Y(PTN)=PY(PTN)
1490   PZ(PTN)=ZI(I)/ZI(M)                    : Z(PTN)=PZ(PTN)
1500  NEXT J
1510 NEXT I
1520 PX(PN)=0 : PY(PN)=0 : PZ(PN)=1
1530 X(PN)=0 : Y(PN)=0 : Z(PN)=1
1540 '
1550 SURN=1
1560 FOR J=0 TO N-2
1570  PL(SURN,0)=4
1580  PL(SURN,1)=1
1590  PL(SURN,2)=J+3
1600  PL(SURN,3)=J+2
1610  PL(SURN,4)=1
1620  SURN=SURN+1
1630 NEXT J
1640 PL(SURN,0)=4
1650 PL(SURN,1)=1
1660 PL(SURN,2)=2
1670 PL(SURN,3)=N+1
1680 PL(SURN,4)=1
1690 SURN=SURN+1
1700 '
1710 FOR I=1 TO M-2
1720  FOR J=0 TO N-2
1730   PL(SURN,0)=5
1740   PL(SURN,1)=N*(I-1)+J+2
1750   PL(SURN,2)=N*(I-1)+J+3
1760   PL(SURN,3)=N*I+J+3
1770   PL(SURN,4)=N*I+J+2
1780   PL(SURN,5)=N*(I-1)+J+2
1790   SURN=SURN+1
1800  NEXT J
1810  PL(SURN,0)=5
1820  PL(SURN,1)=N*(I-1)+N+1
1830  PL(SURN,2)=N*(I-1)+2
1840  PL(SURN,3)=N*I+2
1850  PL(SURN,4)=N*I+N+1
1860  PL(SURN,5)=N*(I-1)+N+1
1870  SURN=SURN+1
1880 NEXT I
1890 FOR J=0 TO N-2
1900  PL(SURN,0)=4
1910  PL(SURN,1)=PN
1920  PL(SURN,2)=N*(M-2)+J+2
1930  PL(SURN,3)=N*(M-2)+J+3
1940  PL(SURN,4)=PN
1950  SURN=SURN+1
1960 NEXT J
1970 PL(SURN,0)=4
1980 PL(SURN,1)=PN
1990 PL(SURN,2)=N*(M-2)+N+1
2000 PL(SURN,3)=N*(M-2)+2
2010 PL(SURN,4)=PN
2020 RETURN
2030 '
2040 LABEL "2DDATA"
2050 M=0
2060 IX=0 : IZ=0
2070 IX0=0 : IZ0=0
2080 LINE (5,0)-(0,0),PSET,2 : LINE -(0,3),PSET,2
2090 CIRCLE (IX/3,IZ/3),.05,7,2.6
2100 '
2110 LOCATE 61,14 : PRINT "8"
2120 LOCATE 59,16 : PRINT "[ UP ]"
2130 LOCATE 50,18 : PRINT "4 [ LEFT]  5   [RIGHT] 6"
2140 LOCATE 59,20 : PRINT "[DOWN]"
2150 LOCATE 61,22 : PRINT "2"
2160 Z$=INKEY$
2170 IF Z$="4" AND IX>0 THEN GOSUB "RESET":IX=IX-1 :GOSUB "PSET"
2180 IF Z$="2" AND IZ>0 THEN GOSUB "RESET":IZ=IZ-1 :GOSUB "PSET"
2190 IF Z$="6" AND IX<15 THEN GOSUB "RESET":IX=IX+1 :GOSUB "PSET
"
2200 IF Z$="8" AND IZ<9 THEN GOSUB "RESET":IZ=IZ+1 :GOSUB "PSET"
2210 IF Z$="5" THEN 2230
2220 GOTO 2160
2230 IF M=0 THEN 2290
2240 XA=R(M-1) : ZA=ZI(M-1)
2250 XB=R(M)   : ZB=ZI(M)
2260 XC=IX     : ZC=IZ
2270 NV=SGN((XB-XA)*(ZC-ZA)-(XC-XA)*(ZB-ZA))    :'VECTOR PRODUCT
2280 IF NV=-1 THEN 2160
2290 CIRCLE (IX/3,IZ/3),.05,7,2.6
2300 LINE (IX0/3,IZ0/3)-(IX/3,IZ/3)
2310 IX0=IX : IZ0=IZ
2320 M=M+1
2330 PRINT M;" R=";IX;" Z=";IZ
2340 R(M)=IX : ZI(M)=IZ
2350 IF IX<>0 THEN 2160
2360 INPUT "DIVISION = ",N
2370 PN=N*(M-1)+2 : PLN=N*M
2380 RETURN
2390 '
2400 LABEL "PSET"
2410 LINE (IX/3-.1,IZ/3)-(IX/3+.1,IZ/3),PSET,7
2420 LINE (IX/3,IZ/3-.1)-(IX/3,IZ/3+.1),PSET,7
2430 LOCATE0,23 : PRINT "R=";IX;"Z=";IZ; :LOCATE 0,M
2440 RETURN
2450 '
2460 LABEL "RESET"
2470 LINE (IX/3-.1,IZ/3)-(IX/3+.1,IZ/3),PSET,-2*(IZ=0)
2480 LINE (IX/3,IZ/3-.1)-(IX/3,IZ/3+.1),PSET,-2*(IX=0)
2490 LOCATE0,23 : PRINT CHR$(5);
2500 RETURN
```

超簡単アニメーション技法
君にもできる“It's a SONY”(?)

動画が描けなくては，どんなに優秀なプログラムがあってもアニメーションは作れません。基本は4コマアニメ。無駄のないスムーズな動きを研究してみましょう。

Fukuhara Toru
福原　徹

C言語ばやりの昨今，かくいう私も最近はMML入力をするときぐらいしかBASICを立ち上げなくなってしまったのです。が，しかし「言語の目的は，そのものを学ぶことではない」という格言（？）もあることですし，今，一番やりたいこと，興味のあることを，最もやりやすい方法でやってみるのがパソコンの主な使い道なのですから，あまり難しく考えずに遊んでみたらいかがでしょうか。たまたま今の私にとってそれがアニメーションだったということです。

少ない画面で最大の効果を

たとえばテレビアニメの雪やモブ(群衆)シーン。さすがに近ごろの劇場版では少なくなりましたが，制作費が非常に不自由なテレビアニメにおいては，少ないセル画をいかに有効に見せるか，ということが重要になってきます。そこで出てくるのが先の雪のシーンなどなわけです。少しヒネクれた（？）目を持っている人ならば，雪の降るシーンに破線のダンスを見たり，突如東京を襲った大怪獣ケゴンから逃げようと必死になっている群衆のなかに，いつまでたっても同じところを走りつづけている変なおじさんを発見したりするのです。

ところが，あの技法というのは思いのほか効果的であって，アラ探しなどをするつもりでなかったら物語自体やその動きの面白さに魅せられてしまって，気になることも少ないはずです。もっとも，そこにはアニメーターたちの長年の努力の結晶がつまっているのであって，にわかアニメーター（？）の私たちにもすぐに同じ技術が使えるかというとそうでもありません。が，逆にいえば，多くの動画を使ったからといって，その苦労や手間に見合った効果が期待できるかというとそうでもないのです。

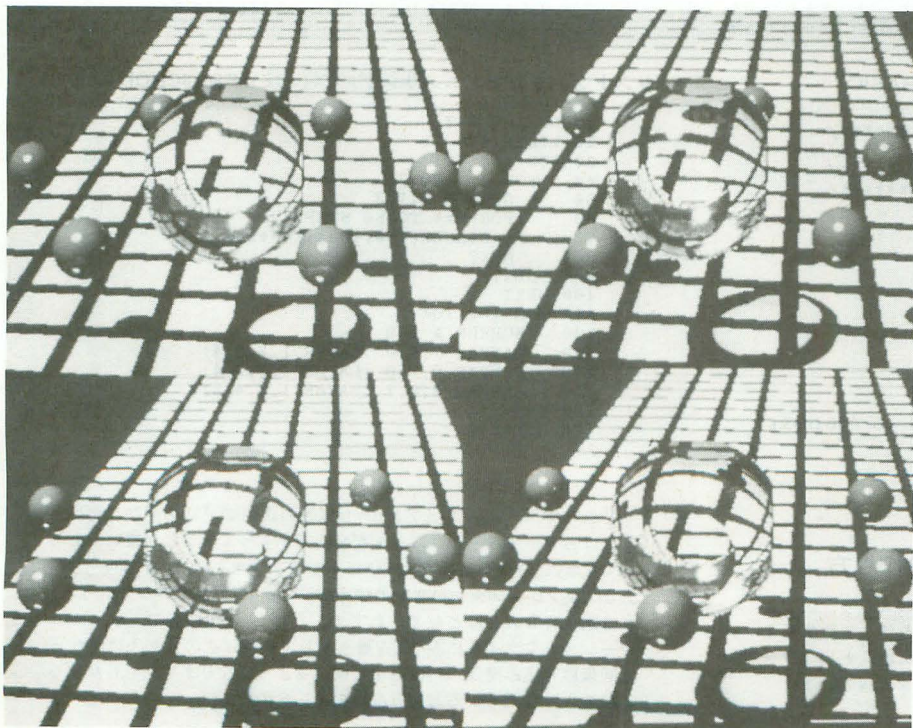
事実，現在の国産アニメのほとんどが「2コマ撮り」「3コマ撮り」といった撮影法を用いています（紙芝居みたいなのさえある）。要はいかに効果的に妥協するか，ということでしょうか。

紙飛行機の楽しさを

コンピュータアニメーションにも同じことがいえるでしょう。機能に制限のあるハードを用いて，いかに多数の動画を動かしているように見せるか。それにはもちろん，プログラミングテクニックも重要でしょう。しかし，むしろちょっとした工夫とアイデアでテクニックを凌ぐ面白いアニメーションが作れるものなのです。

創造的な仕事をする際によくいわれることに，「技術や絵心はなくたって，よいアイデアさえあれば！」というのがあります。これはなにもアニメに限ったことではないのですが，でも本当にそうかなぁと疑問に思う人も多いでしょう。

しかし，そのひとつの答えとして子供たちの作った作品群があります。彼らはおおよそテクニックや技術などという言葉からは縁遠いけれども，そのやわらかな感性と湧き出す想像力，そして豊かな遊び心で，時として大人の心さえも動かす作品を作り上げることさえあるのです。正直いって，彼らの創造性に私たちはかないません。しかし，それを見習うことは私たちの作品制作に多くのヒントを与えてくれるのではないでしょうか。

どうやらBASICなどの言語にもそれはあてはまるようで，近ごろの高度な市販ソフトに毒された私たちの感覚には，日曜プログラミングがなんだか恥ずかしくて労力の無駄遣いみたいに感じてしまうようです（思わないとは言わせないぞ。じゃなかったらもっとピコピコゲームが増えてもいいはずだ)。でも本当はそんなことはないはずですよね。F-14飛ばすのも気持ちいいかもしれないけれど，ラジコン飛行機を作って飛ばす楽しみ，いや紙飛行機を折るのだって人間の創造意欲を刺激してくれるものなのです。

さて紙飛行機プログラミングにはやはりBASICが最適ではないでしょうか。書いたら即実行に移れるのってインタプリタの利点だけど，私のようにCばかりやってるとその点がやけに便利に感じてしまうのが面白いところです。

まず今回の私のアプローチを。最初に，「アニメが作りたい！」という衝動が起きたわけです(当たり前ですね)。でもスプライトは使いたくなかった。なぜってあれはただのセル引きであって本当の意味でのアニメとは私には思えなかったからです。どうせやるなら動画を使いたい，しかもよくあるワイヤフレームじゃなくて，フルカラー，フルスクリーンのグラフィックで……。

そうなると，リアルタイムで絵を描かせるわけにはいかなくなってしまうし，第一そんな絵を描く能力を私は持っていない。よし！　こうなったら贅沢にレイトレツールを利用してしまおう。これなら引数をちょいといじれば見栄えのする動画が描ける。しかも精密に。そしてその絵を高速に切り換えればいい。その程度のプログラムならBASICで十分だ……。つまりこの記事の主要な部分はレイトレツールに寄りかかっちゃっているのですが，まあBASICというのは目的ではなくて道具なのだよと善きに解釈して，笑って許してやってください。

クルクル回るレイトレ画面

と，いうわけで，超単純なサンプルを用意してみました。これは4枚のレイトレーシング画面をループ状に繰り返し表示するプログラムです。まず表示画面256×256，実画面512×512，65536色モードのスクリーン上に，画面を4分割した形に並べた動画を読み込みます。そしてその表示画面位置をhome関数を使って変えているわけです。70～80行はその座標値を格納している2行2列の配列です。

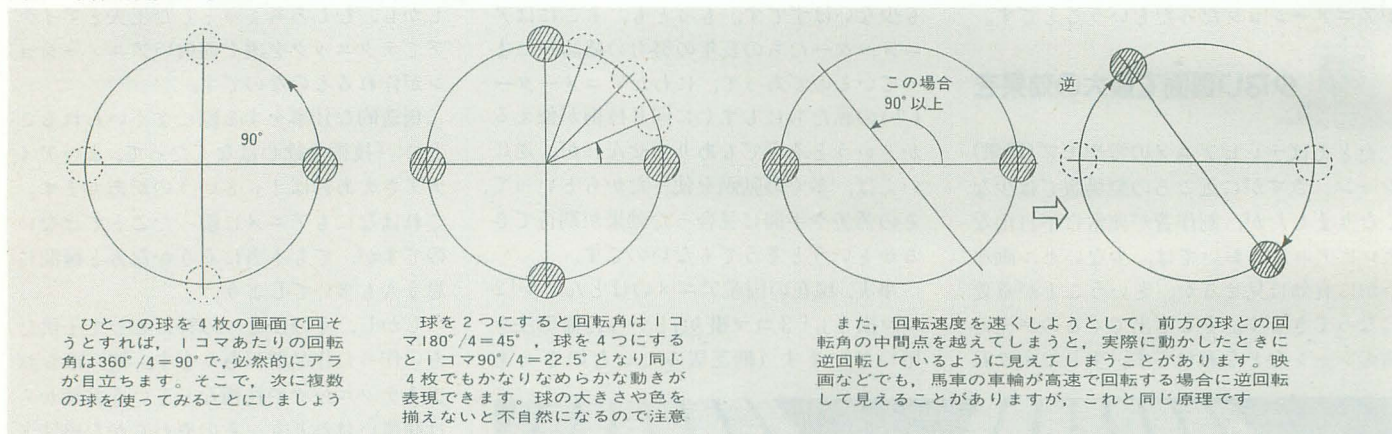
「なにぃ！　たった4枚？」などと言わずに，とりあえず試してみてください。馬鹿にするほどひどいものではないと思いますよ。ちょっとした環境ビデオくらいにはなるでしょ。今回は正確な原画の作りよさからC-TRACE68を使ってみましたが，その気になって，Z'sSTAFF PRO-68Kを使って描いたイラストなり，3DのCADデータなりを用いれば，あなただけのオリジナルアニメーションの出来上がりというわけです。

このサンプルを見て，「結構いけるじゃないか」と思ったら，自分でもっと面白い絵を描いて楽しめばいいし，「4枚なんてバカバカしくってやってらんないよ」と思ったなら8枚，16枚，32枚と動画を増やしていけばよいのです（もっとも4枚で満足するようでは困るのですが)。そういえば最近発売された「彩CRONE」は，ローカル座標系

図1　回転運動のちょっとしたテクニックと注意

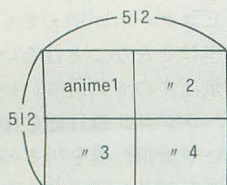
ひとつの球を4枚の画面で回そうとすれば，1コマあたりの回転角は360°/4＝90°で，必然的にアラが目立ちます。そこで，次に複数の球を使ってみることにしましょう

球を2つにすると回転角は1コマ180°/4＝45°，球を4つにすると1コマ90°/4＝22.5°となり同じ4枚でもかなりなめらかな動きが表現できます。球の大きさや色を揃えないと不自然になるので注意

また，回転速度を速くしようとして，前方の球との回転角の中間点を越えてしまうと，実際に動かしたときに逆回転しているように見えてしまうことがあります。映画などでも，馬車の車輪が高速で回転する場合に逆回転して見えることがありますが，これと同じ原理です

リスト1　4コマアニメサンプル(X-BASIC)

```
 10 /*******************************************/
 20 /*        a n i m a t i o n               */
 30 /*          s a m p l e       ver 1.00    */
 40 /*******************************************/
 50 str na="genga"
 60 int i,c
 70 dim int x(1,3)={  0,256,256,   0,
 80                   0,   0,256,256 }
 90 screen 1,3,1,1
100 vpage(0)
110 img_load(na,0,0)
120 img_scrn(0,3,1)
130 vpage(1)
140 while 1
150    for c=0 to 3
160       home(0,x(0,c),x(1,c)):for i=0 to 600:next
170    next
180 endwhile
```

リスト2　6コマアニメサンプル(X1 CZ-8F80/8CBOI Ver1.0)

```
10 DEFINT A-Z
20 WIDTH 40
30 FOR I=0 TO1
40   FOR J=1 TO 3
50     SCREEN I,I,J
60     FOR K=0 TO180 STEP 6
70       CIRCLE(160,100),K+J+I*3,1,1
80     NEXT
90   NEXT
100 NEXT
110 FOR I=0 TO1
120   SCREEN I,I,0
130   CANVAS 7,0,0 :LAYER 1,2,3,4:
140   CANVAS 0,7,0 :LAYER 1,3,2,4:
150   CANVAS 0,0,7 :LAYER 1,3,4,2:
160 NEXT
170 GOTO110
```

画像データの作り方

4枚の動画を描き終えたら，図のような配置にFUT.Xを使って編集します。それを“Genga.gl3”というファイル名で画像データに落とします

ところで，BASICマニュアルのimg関数の説明に，「この関数を実行するにはカラーイメージユニットが必要です」ってなことが書いてあるのですが,このワンセンテンスにだまされて「img_save( )やimg_load( )はオプションを付けないと使えないのか」などと思わないように。私は購入後3カ月ほどだまされてましたけど。BASICのコンフィギュレーションファイルにIMAGEを書き足してやれば次の立ち上げ時からは画像取り込み関係以外は自由に使えるようになります。もしかすると私のような人もいるといけないので，念のため。

を取り入れているそうですから，アニメーションを作りたい人にとってはびしばし使えるツールになるかもしれませんね。これからはCGアニメーションもどんどん身近になってくることでしょう。

＊　＊　＊

この道も極めようとすれば，それなりの苦労が伴います。しかしやってみないうちから諦めてしまっては進歩もなにもありません。まずは遊びのつもりで何事も試してみてください。そしてそのなかで築いた技術や，感じたことを次の挑戦に生かしていくのです。自分の描いた絵が動くのを見るのは，たとえそれが「へのへのもへじ」であっても嬉しいものなのですよ。

〈参考〉 C-TRACE68による動画の作成

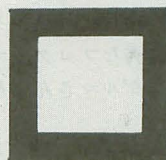
リスト3-a C-TRACE68用の元絵データ “anime. rdt”

```
 1: /*----------------------------------------
 2:       file_name "anime.rdt"
 3:       Animation sample    for C_Trace68
 4:         後ろに注釈文があるところは、購
 5:       画番号1 2 3 4それぞれについて、
 6:       さしかえて下さい。
 7:   ----------------------------------------*/
 8: ratt
 9: clear  0  0  0 0.2 0.2 0.2 10 100
10: red    1  0  0 0.2 0.2 0.2 60 100
11: haikei 0  0  0 0.3 0.3 0.3 20 100
12:         :cmap masu 1.3 pl    0    0  40  40
13:                          -200 -1100 200 1100
14:         :mxyz 0 -100  0  /*-- "mz"  のパラメータを 0 -10 -20 -30 --*/
15: end
16:
17: tatt
18: glass  1  1  1  1.52
19: mud    0  0  0  1
20: end
21:
22: shp
23: diamond1 bx 30 30 30
24: diamond2 sp 35
25: dai      bx 150 5 1000 :my -100 *r haikei *t mud
26:                                        /*----------------------*/

27: hoshi1   sp 10 :mx 80 :ry   0  *r red *t mud  /*ry=   0  15  30  45 */
28: hoshi2   sp 10 :mx 80 :ry  60  *r red *t mud  /*ry=  60  75  90 105 */

29: hoshi3   sp 10 :mx 80 :ry 120  *r red *t mud  /*ry= 120 135 150 165 */
30: hoshi4   sp 10 :mx 80 :ry 180  *r red *t mud  /*ry= 180 195 210 225 */
31: hoshi5   sp 10 :mx 80 :ry 240  *r red *t mud  /*ry= 240 255 270 285 */
32: hoshi6   sp 10 :mx 80 :ry 300  *r red *t mud  /*ry= 300 315 330 345 */
33: end                                       /*----------------------*/
34:
35: log
36: dia     :* diamond1 diamond2
37:         *r clear *t glass
38: end
39:
40: lit
41: taiyou pa 0.3  -1  -0.5 10000 1 1 1
42: end
43:
44: back 0  0 .4
45:
46: allend
```

リスト3-b C-TRACE68のKey File “anime. key”

```
 1: vi      40 130  340
 2: ta      0   -8    0
 3: sc      0    1    0
 4: zo      28
 5:
 6: m a     1 2 2 0
 7:
 8: st      1   1
 9:
10: ar      255 0 511 255
11:
12: ra      1.33333
13:
14: re      anime
15:
16: output 1 1 anime1
17:         /*  anime1 anime2 anime3 anime4 */
18: ru
```

カラーマッピングデータについて

Z'sSTAFF PRO-68Kなどのグラフィックツールを使い，図のような絵を描いておき（好きなデザインでよい），FUT.Xを使って“masu.img”というファイル名でコンバートしておいてください。それを同じディレクトリに置いてからRAY.Xを起動してください

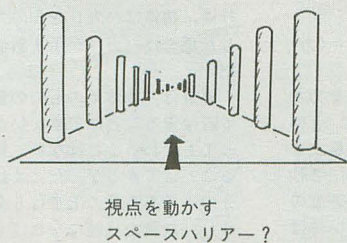
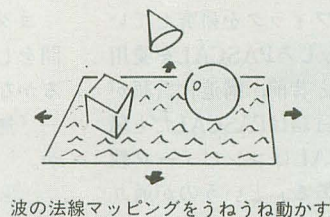
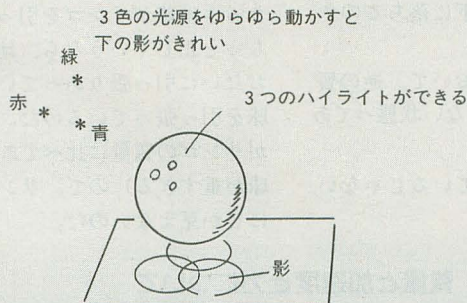
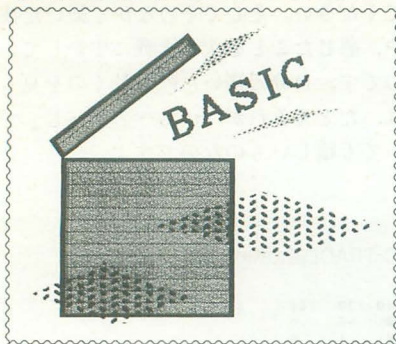
さまざまな応用例

3色の光源をゆらゆら動かすと
下の影がきれい

波の法線マッピングをうねうね動かす

視点を動かす
スペースハリアー？

お星さまをぐるぐる回す

友達の顔をカメラで
画像取り込み

手軽に重力シミュレーション

永遠に落ち続けるリンゴの話

難しいものの代名詞的な「物理学」ですが，徹底的に数式化されている物理の世界はパソコンには扱いやすいものです。BASICからシミュレーションしてみましょう。

Tan Akihiko
丹 明彦

「FORTRANは，科学技術計算にもっとも適した言語である」と，いったい誰がいいだしたんだろう。確かにこの言葉は間違ってはいないが，こういったほうが正確だと思う。「FORTRANは科学技術計算にもっとも広く使われている言語である」。そう，物理・化学などの自然科学界では，研究に使う計算機の共通言語といえば，まずFORTRANである。反対にFORTRANを知らなければ，どんなに優秀なプログラマといえどもたちまちにしてダルマさんになってしまうのだ。

このテのことはどの世界にもあることなのだろうが，コンピュータ界にはとりわけたくさんあるような気がする。物理屋さんや化学屋さんがFORTRANを愛用している（もしくは愛用しなければならない）ことは先にいったとおりだが，計算機屋さん，つまりコンピュータそのものの研究をなりわいとしている人々の間では，FORTRANは必ずしももてはやされているわけではない。まあいちばん大きな顔をしているのは，たぶんC言語だろう。「Cはシステム開発言語である」なんてこともよくいわれる。

コンピュータグラフィックを研究している先生方はCよりもむしろPASCALを愛用しているそうだ。ひと昔前，構造化言語が脚光を浴び，次世代言語はPASCALだと騒がれた頃は，「PASCALはコンピュータ教育用に適した言語である」というのが通り相場だった。それっきりPASCALは消えてしまったとばかり思っていたのだが，結構意外なところで使われている。

だいぶ話がそれてしまったので要点をいうが，数値計算は決してFORTRANの専売特許ではないと思うのだ。そんなことをいっていたら，BASICしか載っていないシステムを使っている人の立つ瀬がない。BASICインタプリタには素晴しい機動性がある。これを生かして，BASICでしかできない楽しい数値計算をやろうではないか。

これから紹介するのは，正確さをモットーとする数値計算ではない。今月は数値演算の特集でもないし，そんなに肩のこるものでは読者も嫌がるに違いない。別にこびたり，へつらったりしているつもりはないが，BASICでやるときくらい，楽しくやりたいものだ。

内容は簡単なシミュレーションだ（残念ながらシミュレーションゲームではない）。簡単ではあるが，物理の基礎のおさらい程度のこともするし，はては微積分の概念にまで迫る予定だ。もしもあなたが理系の受験生なら，ひょっとしたら物理や数学の理解を少しでも助けてくれるところがあるかも。

ものはなぜ落ちる

「ものはなぜ下に落ちる（この言葉は縁起が悪いかな）の？」
—上に落ちたら変じゃない？　下だから，「落ちる」っていうんだろう？
「……」
—へへへ，冗談だよ。本当はこうだろ。ニュートンのおかげでものは下に落ちるのさ。どうだ，正解だろ？

ヨタはこのくらいにしておいて，逆の質問をしよう。ものが落っこちない状態ってあるかな？
—「無重力状態」に決まっているじゃないか。

そうだね。この言葉に，最初の質問のヒントがありそうだ。「無重力状態」は，「重力」が「無」い「状態」と読める。物体が落ちる原因は，実はこの「重力」なのだ。

重力，引力ともいうが，その正体は，地球という大きな物体がリンゴを自分の中心に引っ張ろうとする力だ。そのためにリンゴは地球の中心に向かって，つまり下へなのだが，力を受け，落ちることになるのだ。

しかし，地球はリンゴとくっついていないはずなのに，どうして引っ張ることができるのだろう。磁石が鉄片を引きつけるように，地球はリンゴを引きつけることができるのだろうか。

この推測はだいたい当たっている。ただし，この場合地球とリンゴとの間に働く力は，「磁力」ではなく，ニュートンが発見した「万有引力」と呼ばれる力だ。

あまり意識されてはいないが，万有引力は，およそ質量を持つ物体なら，どんなものにも存在する。机の上にボールを2つ置いたら，その間にも万有引力は働いている。さっき地球がリンゴを引っ張るといったが，もっと正しくいうなら，地球とリンゴは，お互いに引っ張りあっている。リンゴも地球を引っ張っているのだ。ただ地球の質量がリンゴの質量に比べてあまりに大きい（地球が重すぎる）ので，リンゴが落ちるようにしか見えないのだ。

質量と加速度と力について

「質量」は「重さ」と似ているが，「重さ」ではない。ニュートンは，「運動方程式」といわれる有名な式を立てている。

$$m \cdot a = F$$

というのがそれで，（質量）×（加速度）＝（力）という関係は非常に簡潔で，美しい。

さて，地球上では，あらゆる物体には重力加速度と呼ばれる加速度gがかかっている。gは定数で，その値は$9.8m/s^2$である。ビルの屋上から放り出された球は，1秒後には9.8m/s，2秒後には19.6m/sの速度で落下する。この速度の増え方は，球の質量には関係ない。重い球は速く落ちるとか，軽い球はゆっくり落ちるとかいうことはない。理由は本文の万有引力の式を見ればわかると思う。

さて質量がmの物体にはF＝m·gという力がかかる。もし，その物体がハカリの上に乗っていれば，物体はハカリをFの力で押さえる。こうした場合にハカリが示す数値がすなわち「重さ」だ。

質量は，物体そのものの量で，ハカリを使って直接量ることはできないが，どこへ持っていっても変わることはない。重さは，ハカリで量ることはできるが，たとえば重力が地球上の6分の1である月の上では6分の1になってしまうし，無重力状態になれば0になってしまう。つまり，重さは重力を抜きにしては考えられない量なのだ。

質量がそれぞれm_1, m_2で, rだけ離れた物体の間に働く万有引力は次の式で表される。万有引力は, 質量が大きいほど, 距離が近いほど強く働く。

$$F=G\cdot m_1\cdot m_2/r^2$$

ここで, Gは万有引力定数と呼ばれる定数で, どんな物体にも同じ値を使う。机の上のボールだろうが, 地球だろうが, 月だろうが, 太陽だろうがだ。ただ, Gの値はひどく小さいので, 少なくとも一方の物体が地球のように巨大な質量を持たないと, Fの値はハカリでは測定できないほど小さい値になってしまう。だから, 万有引力の存在は, ふだん意識されることはない。電車に乗っているあなたの隣にちょっと太めの人が座っているからといって, 引力を感じることなんてないはずだ。

例として, 地球上の物体に働く万有引力を調べてみよう。地球の質量をM, 物体の質量をm, 地球の半径をRとすれば,

$$F=G\cdot M\cdot m/R^2$$
$$=m\cdot (G\cdot M/R^2)$$

となる。これを運動方程式 (カコミ参照),

$$F=m\cdot a$$

と比較すると,

$$a=G\cdot M/R^2$$

となり, 加速度が得られる。これが地表での重力加速度gである。GもMもRも, 地球に関する定数だから, 物体の質量とはなんの関係もないことがわかる。

積分の概念

以上のことから, 2つの物体を置いたとき, その間に働く力が求められ, その値を物体の質量で割ってやれば, それぞれの加速度がわかる。しかし加速度は目に見える量ではない。目に見える量は速度や座標だけだ。与えられた情報は加速度だけだ。なんとかして速度や座標を取り出せないものか。

ちょっとでも物理をやった人なら, 次のような式を呪文のように丸暗記したことがあるのではないだろうか。

$$\int(\text{加速度})dt=(\text{速度})$$
$$\int(\text{速度})dt\ =(\text{距離})$$

すなわち, 加速度は1秒当たりの速度の増え高なので, 加速度を「時間について積分」すると速度になり, 速度は1秒当たりの距離の変化だから, 速度を「時間について積分」すると距離になるという考え方だ。しかし,「時間について積分」という考え方は結構馴染みにくい。長さを積分すると面積, 面積を積分すると体積になるというのはみんな知っているだろうが。ここで少し脱線して, 積分の概念について一席ぶってみよう。

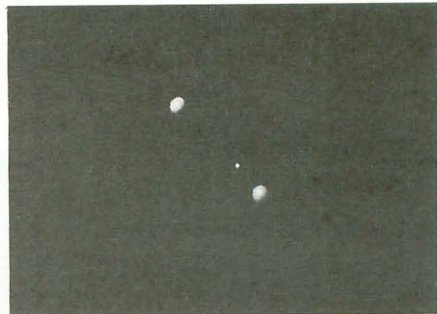
微分積分といえば, 数学をあまりよく知らない人からも恐れられているが, これはなにも, 文部省が受験生を苦しめるために用意したものではない。微積分学もまた, ニュートンが始めたと記憶しているが, もともとは物理学の問題を解決するための一手法として導入されたのだ。

簡単なところで説明しよう。毎秒10メートルの速度で進んでいる車は, 1秒間にどれだけ進むだろうか。そんなの10メートルに決まっているじゃないか, なんていわずに, なぜそうなるのか考えてみよう。たとえばその1秒間を0.1秒ごとに区切り, 10個の区間に分けてみよう。初めの区間で, 車は10(メートル毎秒)×0.1(秒)=1 (メートル) 進む。次の区間でも1メートル進む。3番目の区間でも…というぐあいに繰り返せば, 1秒間に進んだ距離は1+1+1+……+1=10(メートル)だと容易にわかる。

これが積分の本質である。この例のように車が一定の速度で進んでいる場合には, 0.1秒などという粗い分け方でもちゃんと正しい答えが出るが, 速度が刻々と変化しているような場合はもっともっと細かく区切っていき, 時間を「無限小」までもっていく。その各区間で無限小の時間に進む距離を計算し, 無限回分を合計する, といったかっこうになる。無限小の時間というのが例の「dt」というやつである。学校で習っていること (不定積分の計算など) は, 無限小だとか無限回だとかいう気の遠くなりそうな計算の手間を省くための計算方法である。だから, 積分は知っておくと便利なものなのだ, と理解しておくと, 別に勉強が嫌なものでもなくなるだろう (そんなことはないかな)。

引力プログラム

たいして正確さが要求されないなら, dtを無限に短い時間と限定しなくても, たとえば0.001秒といった十分に短い時間に設定すれば, 積分でない単なる進んだ距離の合計でも, 結構誤差は抑えられる。これは嘘でもハッタリでもない。これから紹介するプログラムで,「積分=微小部分の足し算」ということを示して見せよう。

プログラムの流れは次のようになる。下準備に2つの物体を置き, 初速を与えておく。それぞれ質量はm_1, m_2, 座標はp_1, p_2, 速度はv_1, v_2としておく。ここからループに入る。

まず物体の座標と質量から, 万有引力がわかる。それぞれ質量m_1, m_2で割ってやれば, 運動方程式 (力÷質量=加速度) から, その瞬間の加速度a_1, a_2が計算できる。それを速度v_1, v_2に足してやれば, 次の瞬間の速度が出る。また, 速度v1, v2を座標p_1, p_2に足してやれば, 次の瞬間の座標も出せる。これの繰り返しなのだ。

本当は, たとえば0.1秒ごとに区切っているのだから, 加速度や速度に0.1を掛けてからそれぞれ速度や座標に足すのが正しいが, このプログラムでは万有引力定数を小さめに設定してあるから, そのまま足しても誤差は少ないので, わかりやすいようにそのまま足している。

なお, 座標と速度と加速度は, ベクトルを使って表示すべきところをx, y成分に分けているので, 万有引力の表式が本文中のものと少し違う (図を参照)。

リスト2は, 質量比が1:2の物体を放り出したときの動きを追っている。実行してみると, はたして2つの球体は楕円軌道を動いている。万有引力に従う物体は楕円軌道を描くことが多い。地球だって太陽の周りを, 非常に正円に近い楕円軌道を描いて回っているのだ。だからこのいい加減な「積分」も, 多少の誤差を我慢すればそれなりに見られるものといえる。

途中, disp2( )とあるが, これは系の重心 (2物体の座標を結ぶ線分を1:2に内分する点) から見た相対的な物体の運動を表示する。2つの球は重心の周りをいつまでも回っている。

disp1( )なら赤い球から見た青い球の運動を表示する。やはり青い球は赤い球の周りを回り続ける。disp3( )では, 少し離れたところから見た動きを観察しているように表示するので, そのうち球はどこかにいってしまう。

これだけでは最初は珍しくてもすぐに飽きてしまうだろうから, ユーザーもシミュレーションに参加できるようにしたのがリスト3である。赤い球はマウスで自由に動

かせる（もちろん画面外には出られない）。青い球は赤い球からの万有引力を受けて動く。うまくマウスで誘導すれば，青い球をいろいろな形の楕円軌道に乗せることができる。

ただ気をつけてもらいたいのは，青い球と赤い球を近づけすぎると危ないということだ。万有引力は，距離が近いほど大きくなるので，近づけすぎると，青い球は大きな万有引力を受ける。したがって速度は大きく変化する。そうなると青い球はあっという間に画面の外に飛び出していってしまう。これは，時間を無限小にとっていないために起こる誤差のひとつだ。気をつけて誘導しよう。どこまで小さい軌道を作れるか頑張ってみるのも面白い。

なお，リスト3では，赤い球と青い球と，それに重心の軌跡を配列にしまっておき，一定の時間がたったら消えるように仕組んである。

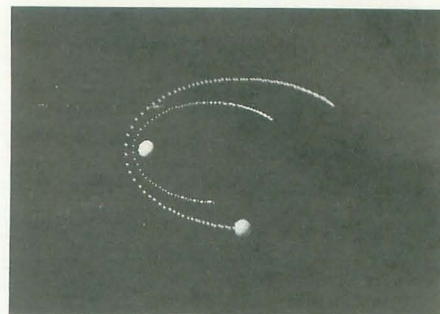
ところで，230行から270行の頭につけている/*を削ると，よりスリリングな展開が楽しめる。周りから緑色の壁がじりじりと迫ってくる。赤か青の球がどちらかでもはみ出すとおしまいである。画面の左上には，ループの回数が表示されているから，どこまでもたせられるかを競ってみてはどうか。壁の迫ってくる速さは調節できる（230行）ので，暇つぶしにゲーム代わりに遊ぶこともできる。

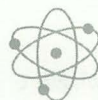
ゲームといえば，これとよく似た動きをするものがあった。ゼビウスのジェミニ誘導がそれだ。あれは速度がありすぎて，僕の手には負えなかったが。ほかにはグラディウスのバリアもちょっと近かったかな。ともかく，ふだん遊んでいるゲームのキャラクターとはひと味もふた味も違う動きをするので，ちょっと新鮮な感じがすると思う。

というのも，ほとんどのゲーム用キャラクターは無慣性運動をする。操作性のために物理法則を無視している。現実には，そんな運動ができるものはありえない。飛び出すな　車は急に止まれない，のである。たまには物理法則にきりきり舞いさせられるようなゲームをやってみたい。ちょっと古いが，「月面着陸ゲーム」みたいなやつを。ロケット噴射しかできない着陸船を適当な位置に着陸させるもので，面が進むと着陸地点の置き方がだんだんイヤラシクなってくるのだ。噴射で速度を微調整しながら，静かに着陸させないと，崖にぶつかったりする。X68000だったらBASICでも結構いいのが書けそうだ。

斥力もできる

自然界には，万有引力とよく似た振る舞いをするものがある。さっきチラッといった磁石の間に働く力がそうだ。電子などの電荷を持った粒子の間に働く力もそうだ。不思議なことだが，どちらも（比例定数は違う）磁気量または電荷の積に比例し，距離の2乗に反比例していて，万有引力とまったく同じ形の式で表される。

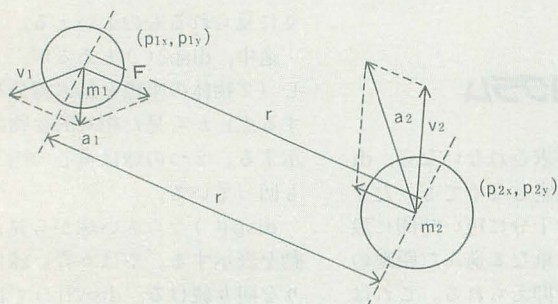
原子と電子について，たいていの人はこういうイメージを持っているだろう。電子は一定の軌道を描きながら，原子の周りをくるくると回っている（このイメージは実は正しくないのだが，説明するのは面倒だし，あくまでイメージだからいいだろう）。これは，太陽と地球，または地球と月の関

係にどことなく似ているような気がしないだろうか。う〜んこの世はフラクタル（本当はこんなところにフラクタルなんて言葉を使ってはいけない）。

電気と磁気が，万有引力と違うのは，万有引力にはその名のとおり「引力」しかないのに対し，反発する力つまり「斥力（せきりょく）」もあるところだ。磁石はN極同士，S極同士は反発するが，N極とS極は引きつけあう。そこで，リスト2の最初のほうの変数gの値の符号を反転して負数にしてみよう。今度は青い球が赤い球から逃げ出す。放っておくとすぐ画面外に出ていくので，誘導は忙しくなる。

球体の計算

今回のプログラムには青い球と赤い球が出てくるが，これにはスプライトを使っている。リスト1は，それを定義するプログラムだが，基本的なアルゴリズムはレイトレーシングなどでやっていることと同じである。本題とはずれるので詳しくは説明しないが，結構リアルな球ができあがる割にはプログラムはごく短いもので済むのである。BASICあなどりがたし，といったところだろう。

実行すると，画面の真ん中に球が出てくる。画面の左肩には，スプライトのパレットが表示されていく。スプライトは，ひとつのパターンに15色までしか使えないので，色減らしをしている。そのため球の色は完全にはグラデーションがかかっていないが，球が小さいせいで，それほど質感は落ちていないだろう。定義が終わると，スプライトのテストに入る。マウスに従って赤い球と青い球が動くので，ちゃんと定義できたかどうか確かめてもらいたい。いいと思ったら，BREAKで実行を中止し，リスト2やリスト3を実行しよう。

実行速度に不満のある人は，コンパイルしてもいいが，実数計算がたくさん出てきているので，数値演算プロセッサをつけないと効果は期待できない。持っていない人

図1　2点間の重力

$$r=(r_x, r_y)=(p_{2x}-p_{1x}, p_{2y}-p_{1y})$$
$$r=|r|=\sqrt{r_x{}^2+r_y{}^2}$$
$$F=F\cdot\frac{r}{|r|}=\left(G\cdot\frac{m_1\cdot m_2}{r^3}\right)r$$
$$=\left(F_x, F_y\right)=\left(G\cdot\frac{m_1\cdot m_2}{r^3}r_x, G\cdot\frac{m_1\cdot m_2}{r^3}r_y\right)$$

はfloat2+.xを使うなどして善処してほしい。

シミュレーションの楽しみ

ともあれ，BASICはひととおりの実数演算はこなせるし，表示機能についてはいろいろと気のきいたこともできる。大型機では味わえない楽しいシミュレーションをいろいろと考えてみてほしい。題材はいくらでもある。ゲームのネタにだってなる。たとえば，「衝突」を忠実にシミュレートするプログラムができれば，ビリヤードができあがる。物理学は，数式の遊びばかりではない。応用のしかた次第で，ゲームにリアリティをつける役目もはたせる重宝なものだということを知っていてほしいものである。

リスト1 ボールの定義 (X-BASIC)

```
10 /****** 球のスプライトパターン定義 ******/
20 /* レンダリング用定数 */
30 int XSIZE=16, YSIZE=16, OX=8: OY=8                  /* スプライトのサイズ・中心 */
40 float radius=8#, ratio=1.33333#                     /* 球の半径・縦横比 */
50 float lx=-0.57735#, ly=-0.57735#, lz=0.57735#       /* 光源の方向 */
60 float R=1#, G=0#, B=0#, I=32#                       /* 赤色・32階調 */
70 float ambient=0.2#, diffuse=0.8#                    /* 環境光・散乱反射係数 */
80 float hlcon=50#, hlpow=10#                          /* ハイライト係数・強度 */
90 /* レンダリング用変数 */
100 int x, y, c, r, g, b                               /* 画面上の座標・色 */
110 float nx, ny, nz                                   /* 球の法線ベクトル */
120 float vx, vy, vz                                   /* 反射方向のベクトル */
130 float xx, yy, zz, z2                               /* 視線と球面の交点 */
140 float dif, hl                                      /* 散乱反射・ハイライト */
150 /* スプライト定義用変数 */
160 dim int pal( 1, 15 )                               /* スプライトパレット */
170 dim char ball( 255 )                               /* スプライトパターン */
180 int n_pal=0, frag, i                               /* パレット定義用 */
190 /* レンダリング */
200 screen 1, 3, 1, 1
210 for y=0 to YSIZE-1
220   yy=y-OY
230   for x=0 to XSIZE-1
240     xx=( x-OX )*ratio                              /* 縦横比補正 */
250     z2=radius*radius-xx*xx-yy*yy                   /* 視線はz軸に平行 */
260     if z2>=0# then {                               /* 交点あるか */
270       zz=sqr( z2 )
280       /* 散乱反射光 */
290       nx=xx/radius: ny=yy/radius: nz=zz/radius     /* 法線は単位ベクトルで */
300       dif=(lx*nx+ly*ny+lz*nz)*diffuse              /* 法線と光線の内積 */
310       if dif<0# then dif=0#
320       dif=dif+ambient                              /* 環境光を加算 */
330       /* ハイライト */
340       vx=2#*nz*nx: vy=2#*nz*ny: vz=2#*nz*nz-1#     /* 反射方向 */
350       hl=lx*vx+ly*vy+lz*vz: if hl<0.2# then hl=0#  /* 反射と光線の内積 */
360       hl=pow( hl, hlcon )*hlpow                    /* べき乗で鋭く */
370       /* 色計算 */
380       r=( R*dif+hl )*I: if r>=I then r=I-1         /* ハイライトは白色 */
390       g=( G*dif+hl )*I: if g>=I then g=I-1
400       b=( B*dif+hl )*I: if b>=I then b=I-1
410       r=r/3*3: g=g/3*3: b=b/3*3                    /* 15色以内に収める */
420       c=rgb( r, g, b )
430       pset( x+256, y+256, c )
440       /* スプライトのパレット */
450       frag=0                                       /* 色の重複がないようにする */
460       for i=1 to n_pal
470         if pal(0,i)=c then {
480           frag=1
490           ball(x+y*16)=i
500           break
510         }
520       next
530       if frag=0 then {
540         n_pal=n_pal+1: ball( x+y*16 )=n_pal
550         pal( 0, n_pal )=c                          /* 赤い球 */
560         pal( 1, n_pal )=rgb( b, g, r )             /* 青い球(色成分を入れ替え)*/
570         print using "###:#####";n_pal, c
580       }
590     }
600   next
610 next
620 /* パレットをセット ***/
630 for j=0 to 1
640   for i=0 to 15
650     sp_color( i, pal( j, i ), j+1 )
660   next
670 next
680 wipe()
690 /* スプライトをテスト */
700 sp_clr()
710 sp_def( 0, ball )
720 sp_disp( 1 )
730 sp_off()
740 sp_on( 0, 1 )
750 while 1
760   mspos( x, y )
770   sp_set( 0, x, y, 256 )                           /* マウスどおりに動く */
780   sp_set( 1, 512-x, 512-y, 512 )                   /* マウスと反対に動く */
790 endwhile
```

リスト2　重力1

```
 10 /****** 2質点の運動 ******/
 20 int st=0                                                  /* ステップ数 */
 30 float p1x=-1#, p1y=0#, p2x=1#, p2y=0#                      /* 座標 */
 40 float v1x=0#, v1y=-0.01#, v2x=0#, v2y=0.01#                /* 速度 */
 50 float a1x=0#, a1y=0#, a2x=0#, a2y=0#                       /* 加速度 */
 60 float m1=1#, m2=2#, g=0.001#                               /* 質量・引力定数 */
 70 float f, fx, fy, r, gx, gy                                 /* 引力・距離・重心 */
 80 screen 1,3,1,1: console ,,0
 90 sp_disp( 1 ): sp_off(): sp_on( 0, 1 )
100 /* メインループ */
110 while( 1 )
120   r=sqr( pow( p1x-p2x, 2# )+pow( p1y-p2y, 2# ) )           /* 距離 */
130   f=g*m1*m2/pow( r, 3 ): fx=f*(p2x-p1x): fy=f*(p2y-p1y)   /* 引力 */
140   a1x=fx/m1: a1y=fy/m1: a2x=-fx/m2: a2y=-fy/m2             /* 加速度＝引力÷質量 */
150   v1x=v1x+a1x: v1y=v1y+a1y: v2x=v2x+a2x: v2y=v2y+a2y       /* 加速度を積分→速度 */
160   p1x=p1x+v1x: p1y=p1y+v1y: p2x=p2x+v2x: p2y=p2y+v2y       /* 速度を積分→座標 */
170   gx=(p1x+p2x*2#)/3#: gy=(p1y+p2y*2#)/3#                   /* 重心 */
180   disp2()
190   st=st+1: locate 0,0: print st                            /* ステップ数表示 */
200 endwhile
210 /* 表示1（相対的な位置）*/
220 func disp1()
230   plot( 0, 0#, 0#, 256 )
240   plot( 1, p2x-p1x, p2y-p1y, 512 )
250 endfunc
260 /* 表示2（重心が中心）*/
270 func disp2()
280   pset( 256, 256, 65535 )
290   plot( 0, p1x-gx, p1y-gy, 256 )
300   plot( 1, p2x-gx, p2y-gy, 512 )
310 endfunc
320 /* 表示3（絶対座標）*/
330 func disp3()
340   pset( 256+gx*75, 256-gy*100, 65535 )
350   plot( 0, p1x, p1y, 256 )
360   plot( 1, p2x, p2y, 512 )
370 endfunc
380 /* 球を表示 */
390 func plot( s;int, x;float, y;float, cd;int )
400   int ix, iy
410   ix=256+x*75: iy=256-y*100
420   sp_set( s, ix+6, iy+8, cd )
430 endfunc
```

リスト3　重力2

```
 10 /****** 質点の誘導 ******/
 20 int st=0, st1, st2, mx, my                                        /* マウス座標 */
 30 int x1=-1, y1=-1, x2=512, y2=512                                  /* 壁の座標 */
 40 int FIFOx(2,99), FIFOy(2,99)                                      /* 軌跡を格納する */
 50 float p1x=-1#, p1y=0#, p2x=1#, p2y=0#
 60 float v1x=0#, v1y=-0.01#, v2x=0#, v2y=0.01#
 70 float a1x=0#, a1y=0#, a2x=0#, a2y=0#
 80 float m1=1#, m2=2#, g=0.001#                                      /* 負なら斥力定数 */
 90 float f, fx, fy, r, gx, gy
100 screen 1,3,1,1: console ,,0
110 sp_disp( 1 ): sp_off(): sp_on( 0, 1 )
120 mouse( 1 ): setmspos( 256+p1x*75, 256-p1y*100 ): mouse( 2 )  /* マウスを定位置に */
130 /* メインループ */
140 while( 1 )
150   mspos( mx, my ): p1x=(mx-256#)/75#: p1y=(my-256#)/(-100#)  /* 質点1はマウスで制御 */
160   r=sqr( pow( p1x-p2x, 2# )+pow( p1y-p2y, 2# ) )
170   f=g*m1*m2/pow( r, 3 ): fx=f*(p2x-p1x): fy=f*(p2y-p1y)
180   a2x=-fx/m2: a2y=-fy/m2                                          /* 質点2のみ影響 */
190   v2x=v2x+a2x: v2y=v2y+a2y
200   p2x=p2x+v2x: p2y=p2y+v2y
210   gx=(p1x+p2x*2#)/3#: gy=(p1y+p2y*2#)/3#
220   st1=st mod 100: st2=(st+1) mod 100                              /* 軌跡の寿命は100ステップ */
230 /*if (st mod 3)=2 then {                                           /* 壁の処理 */
240 /*  box( x1, y1, x2, y2, 0 )
250 /*  x1=x1+1: y1=y1+1: x2=x2-1: y2=y2-1
260 /*  box( x1, y1, x2, y2, 63488 )
270 /*}
280   if plot01( 0, p1x, p1y, 256, 1984 )=1 then end                  /* はみ出したら終わり */
290   if plot01( 1, p2x, p2y, 512, 62 )=1 then end
300   plot2( gx,  gy, 21140 )
310   st=st+1: locate 0,0: print st
320 endwhile
330 /* 質点1（赤い球）・質点2（青い球）の表示 */
340 func int plot01( s;int, x;float, y;float, cd;int, col;int )
350   int ix, iy
360   ix=256+x*75: iy=256-y*100
370   if ( ix<x1 or iy<y1 or ix>x2 or iy>y2 ) then return( 1 )  /* はみ出したら */
380   sp_set( s, ix+6, iy+8, cd )
390   FIFOx(s,st1)=ix: FIFOy(s,st1)=iy                               /* 軌跡をセット */
400   pset( FIFOx(s,st2), FIFOy(s,st2), 0 )                          /* 古い点は消す */
410   pset( FIFOx(s,st1), FIFOy(s,st1), col )                        /* 新しい点を打つ */
420   return( 0 )
430 endfunc
440 /* 重心の表示 */
450 func plot2( x;float, y;float, col;int )
460   int ix, iy
470   ix=256+x*75: iy=256-y*100
480   FIFOx(2,st1)=ix: FIFOy(2,st1)=iy
490   pset( FIFOx(2,st2), FIFOy(2,st2), 0 )
500   pset( FIFOx(2,st1), FIFOy(2,st1), col )
510 endfunc
```

X1/X1turbo用

コミカルロボットゲーム TAMA

Iwasaki Naoaki 岩崎 直明

卵型のキャラクターとピコピコっとした動きがかわいい，ロボットバトルシミュレーションゲーム（？）です。オリジナルチューンアップして最強のTAMAを作り出してください。

そこには戦いがあった

かつてその星には高度な文明が存在していた。ありがちな話だが，高度に発達した科学技術は恐るべき最終兵器をも生み出し，自らの手でその母体となった文明そのものを破壊し尽くしてしまったのだ。しかし，戦いは終わっていなかった。世界を焼き尽くした最終兵器とその自動生産工場がいまなお破壊活動を繰り返しているのだ。実に迷惑な話である。

その最終兵器とはなにか？　その名も“Tiny Automatic Mobile Armor”，略してTAMA。彼らの武器は頭突きと火炎である。どうだまいったか。

まずは入力

このゲームはBASICで作られたバトルシミュレーションです。リストはturboBASICですのでX1turboユーザーの方は，BASIC CZ-8FB02/03を起動してそのまま入力してください。X1ユーザーの方はCZ-8FB01V1.0を起動してリストを入力し，170行の，

WIDTH 80,25,0,0

を，

WIDTH 80

に，540行を，

IF INKEY$="" THEN 540

に変更し，30行のKEYLIST，1970行のCBLACK，1970，2070，2100，2910行のKMODEを削除，そのほか460～530，550,4730～4910行を削除してください（そのほかの行番号は変更しないように)。

栄光のバトルフィールドへ

プログラムを起動するとタイトルデモが始まります。なにかキーを押してください(キー反応が悪いのは単なる手抜きです)。メニュー画面が出ましたね。あとはメニューに従って，1から6のキーで処理を選んでいくだけです。

1)　GAME

TAMA同士を戦わせるモードです。PLAYER1のTAMA（左側）とPLAYER2のTAMA の番号を指定してください。画面が切り換わり戦闘モードになります。両者の体力は棒グラフで示されています。これがなくなると負けてしまいます。

2)　LOOK

登録されているTAMAのデータを見ることができます。見たいTAMAの番号を入力してください。各パラメータが表示されます。

3)　EDIT

新しいTAMAを作ることができます。まず名前を入力してください。名前は12文字までで，X1turboでは漢字を使うこともできます。次にロボットに与えるパラメータを決定します。パラメータの内容については次項を参照してください。キー操作はテンキーの2，4，6，8で選択，リターンキーで決定となっています。TAMAは64台まで登録できるようになっています。

4)　KILL

登録されているTAMAをメモリ上から削除します。削除したいTAMAの番号を入力してください。ディスク上に書かれているデータはセーブを行わないと削除や書き換えは行われません。

5)　SAVE

登録されているすべてのTAMAのデータをディスクに登録します。テープには対応していないので注意してください。

6)　END

お疲れさまでした。ゲームを終了してBASICに戻ります。

このゲームで行う操作は以上のとおりです。あとはいかに強いTAMAを作りあげるかにかかっています。

最強のTAMAを作れ

TAMAのパラメータは4つだけです。すなわち，

SPEED	移動速度
DEFEND	攻撃回避能力
ATTACK	戦闘能力
WEAPON	火力

です。

プレイヤーはTAMAをパワーアップするために割り振りできるポイントを50持っています。これを各パラメータの最低値が1以上になるように，4つのパラメータに配分していってください。それぞれのパラメータには以下のような特徴があります。

1)　SPEEDが高ければ相手は狙いが定まらず火を吐きにくくなります。

2)　DEFENDが高くなれば，相手の攻撃が当たりにくくなります。ただし，攻撃が当たったときのダメージには関係ありません。

3)　ATTACKが高ければ，頭突き，火炎による攻撃が当たりやすくなります。また，頭突きによって与えることのできるダメージが大きくなります。

4)　WEAPONが高くなれば，火炎で与えるダメージが大きくなります。

戦闘中はただ黙ってなりゆきを見守るしかないので，これらのパラメータのバランスと運だけが勝負の鍵です。サンプルデータを以下に示します。

MAPO S= 7 D=20 A=22 W= 1

ITA-mk3 S=10 D=15 A=20 W= 5

TAMAGON S= 5 D= 8 A=20 W=17

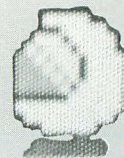
実はこのゲーム，3年ほど前にポケコンで作ったゲームのパワーアップ版なのです。といっても，メイン部分はまったく同じで以前のデータが使えるようにし，その代わりにエディタの部分と演出に凝ってみました。しかも，すべてのTAMAのデータをまとめて管理しようとしたために，たいした内容もないのに長いプログラムになってしまいました。どうかこのTAMAを皆さんかわいがってやってください。

Profile

◇岩崎さんは山形県にお住まいの20歳，現在大学2年生です。マイコン歴2年のXIturboZユーザーで，山形大学のパソコンサークルFLOPSのメンバーです。

リスト TAMA

```
10 ' Tiny Automatic Mobile Armor
20 ' --- INIT ---
30 CONSOLE:SCREEN0:CLS4:TEMPO7500:KEYLIST0
40 CLEAR:DIM DT(3,1),HP(1), X(1),K(1),I(1),FA$(7),TA$(7),CH$(1,15)
50 DEFINT A-Z
60 ON ERROR GOTO "ERROR"
70 GOSUB "PCGSET"
80 PRINT"IPL is not loading T·A·M·A"
90 ' --- LOAD DATA ---
100 OPEN "I",#1,"TAMA DATA.Dat"
110 FOR I=0 TO 7
120  INPUT #1,FA$(I)
130  TA$(I)=MID$(FA$(I),2)
140 NEXT I
150 CLOSE#1
160 ' --- TITLE ---
170 WIDTH 40,25,0,0:INIT:CLS:CGEN1:KMODE0
180 FOR K=2 TO 29 STEP 9
190  CONSOLE:CONSOLE0,24,K,8:CLS
200  FOR I=0 TO 7:READ D
210   LOCATE K,1:PRINT RIGHT$("0000000"+BIN$(D),7)
220   LOCATE K,0 :PRINTCHR$(15)
230  NEXT I
240  FOR J=0 TO 10
250   LOCATE K,0:PRINTCHR$(15)
260  NEXT J
270  PLAY"O7A5"
280  FOR M=2 TO 4
290   MM=(M^1.5)
300   FOR L=0 TO 9-MM
310    LOCATE K,22:PRINTCHR$(14)
320   NEXT L
330   FOR P=0 TO 50:NEXT P
340   FOR J=0 TO 9-MM
350    LOCATE K,0:PRINTCHR$(15)
360   NEXT J
370   PLAY"O7AAA"
380  NEXT M
390 NEXT K
400 CGEN0:CONSOLE
410 PAUSE 10
420 I=0
430 LOCATE 14,8:COLOR 1:PRINT "Hit any key.":PAUSE1
440 LOCATE 14,8:COLOR 5:PRINT "Hit any key.":PAUSE1
450 LOCATE 14,8:COLOR 7:PRINT "Hit any key."
460 ' ---- MUSIC ----
470 TEMPO 220
480 SOUND11,10 :SOUND12,50 :SOUND13,0
490 RESTORE 4740
500 READ A$,B$,C$ :IF A$="0" THEN 490
510 IF A$="0" THEN 510
520 PLAY "V16:V14:V14"
530 PLAY@ A$+":"+B$+":"+C$ :I=0
540 IF INKEY$<>"" THEN 560
550 IF I<200 THEN I=I+1:GOTO 540 ELSE 500
560 LOCATE 14,8:PRINT "Just wait.   "
570 ' --- PRINT 1 ---
580 LABEL "PRINT1"
590 TEMPO 7500:SCREEN0:COLOR,0:CSIZE2:CLS4:WIDTH 80:S=1
600 LOCATE 2,1:COLOR 6:PRINT#0 "T";:COLOR 7:PRINT#0 "iny"
610 LOCATE 2,2:COLOR 6:PRINT#0 "A";:COLOR 7:PRINT#0 "utomatic"
620 LOCATE 2,3:COLOR 6:PRINT#0 "M";:COLOR 7:PRINT#0 "obile"
630 LOCATE 2,4:COLOR 6:PRINT#0 "A";:COLOR 7:PRINT#0 "rmor"
640 CSIZE0
650 FOR I=0 TO 3
660  LINE (4+I*160,52)-(156+I*160,187),PSET,5,B
670 NEXT I
680 LINE (164,4)-(636,44),PSET,5,B
690 ' --- PRINT 2 ---
700 LABEL "PRINT2"
710 CONSOLE:KMODE1
720 F$=""
730 FOR J=0 TO 3
740  F$=F$+MID$(TA$(J),S*3-2,3)
750 NEXT J
760 IF S>=65 THEN "MENU"
770 LOCATE ((S-1)¥16)*20+1,S-((S-1)¥16)*16+6
780 IF F$="" THEN PRINT SPACE$(16):GOTO "MENU"
790 PRINT " "+RIGHT$("0"+STR$(S),2)+" "+F$
800 S=S+1
810 GOTO 720
820 ' --- MENU ---
830 LABEL "MENU"
840 NO=0:CONSOLE 1,4,22,57:CLS
```

```
850 PRINT " ** MENU **"
860 PRINT "(1)-GAME (2)-LOOK (3)-EDIT (4)-KILL (5)-SAVE (6)-END"
870 K$=INKEY$:IF K$<"1" OR  K$>"6" OR K$="" THEN 870
880 ON VAL(K$) GOTO "CALL","LOOK","EDIT","KILL","SAVE","END"
890 ' --- LOOK ---
900 LABEL "LOOK"
910 CLS:PRINT " ** LOOK **"
920 INPUT" No.=";NO
930 NA$(I)=""
940 IF NO<1 OR NO>S-1 THEN 900
950 FOR J=0 TO 3
960  NA$(I)=NA$(I)+MID$(TA$(J),NO*3-2,3)
970 NEXT J
980 PRINT "  "+NA$(I)
990 PRINT " SPEED =";VAL(MID$(TA$(4),NO*3-2,3));
1000 PRINT "  DEFEND=";VAL(MID$(TA$(5),NO*3-2,3));
1010 PRINT "  ATTACK=";VAL(MID$(TA$(6),NO*3-2,3));
1020 PRINT "  WEAPON=";VAL(MID$(TA$(7),NO*3-2,3));
1030 WHILE INKEY$="":WEND
1040 GOTO "MENU"
1050 ' --- EDIT ---
1060 LABEL "EDIT"
1070 IF S>=65 THEN "PRINT2"
1080 CLS:PRINT " ** EDIT **"
1090 PT(0)=1:PT(1)=1:PT(2)=1:PT(3)=1
1100 NA$="":INPUT "NAME=";NA$:NA$=NA$+SPACE$(12)
1110 IF NA$=SPACE$(12) THEN "PRINT2"
1120 LOCATE 41,1:PRINT "    [8]     SPEED = 1  LEFT=46"
1130 LOCATE 41,2:PRINT " [4] + [6]  DEFEND= 1"
1140 LOCATE 41,3:PRINT "    [2]     ATTACK= 1"
1150 LOCATE 41,4:PRINT " [RETRN]    WEAPON= 1";
1160 SY=(SY+(K$="8")-(K$="2")) AND 3
1170 PT(SY)=PT(SY)+(K$="4")-(K$="6")
1180 ST=PT(0)+PT(1)+PT(2)+PT(3)-PT(SY)
1190 PT(SY)=PT(SY)+(ST+PT(SY)>50)-(PT(SY)<=0)
1200 ST=ST+PT(SY)
1210 IFK$=CHR$(13) THEN 1280
1220 LOCATE 60,SYY+1:COLOR 7:PRINT RIGHT$("0"+STR$(PT(SYY)),2);
1230 LOCATE 60,SY+1:COLOR 6:PRINT RIGHT$("0"+STR$(PT(SY)),2);
1240 LOCATE 69,1:COLOR 5:PRINT RIGHT$("0"+STR$(50-ST),2);
1250 SYY=SY
1260 K$=INKEY$:IFK$="" THEN 1260
1270 GOTO 1160
1280 COLOR 7:LOCATE 22,4:PRINT "Are you sure? (Y/N)";
1290 K$=INKEY$:IF K$="" THEN 1290
1300 IF K$<>"Y" AND K$<>"y" THEN "MENU"
1310 FOR I=0 TO 3
1320  TA$(I)=TA$(I)+MID$(NA$,I*3+1,3)
1330 NEXT I
1340 FOR I=4 TO 7
1350  TA$(I)=TA$(I)+RIGHT$("00"+STR$(PT(I-4)),3)
1360 NEXT I
1370 GOTO"PRINT2"
1380 ' --- KILL ---
1390 LABEL "KILL"
1400 CLS:PRINT "  ** KILL **"
1410 INPUT" No.=";NO
1420 F$=""
1430 IF NO<1 OR NO>S-1 THEN "MENU"
1440 FOR J=0 TO 3
1450  F$=F$+MID$(TA$(J),NO*3-2,3)
1460 NEXT J
1470 LOCATE 36,2:PRINT F$
1480 PRINT "Are you sure? (Y/N)"
1490 K$=INKEY$:IF K$="" THEN 1490
1500 IF K$<>"Y" AND K$<>"y" THEN "MENU"
1510 FOR I=0 TO 7
1520  TA$(I)=LEFT$(TA$(I),NO*3-3)+MID$(TA$(I),NO*3+1)
1530 NEXT I
1540 S=NO
1550 GOTO "PRINT2"
1560 ' --- SAVE ---
1570 LABEL "SAVE"
1580 CLS:PRINT " ** SAVE **"
1590 PRINT "Are you sure? (Y/N)"
1600 K$=INKEY$:IF K$="" THEN 1600
1610 IF K$<>"Y" AND K$<>"y" THEN "MENU"
1620 PRINT " Now saving..."
1630 OPEN "O",#1,"TAMA DATA.Dat"
1640 FOR I=0 TO 7
1650  FA$(I)=STR$(I)+TA$(I)
1660  PRINT #1,FA$(I)
1670 NEXT I
1680 CLOSE#1
1690 GOTO "MENU"
1700 ' --- CALL ---
1710 LABEL "CALL"
1720 CLS:PRINT " ** CALL **"
1730 FOR I=0 TO 1
1740  LOCATE22,2+I:PRINT "PLAYER";I+1;
1750  INPUT" No.=";NO
1760  NA$(I)=""
1770  IF NO<1 OR NO>S-1 THEN "MENU"
1780  FOR J=0 TO 3
1790   NA$(I)=NA$(I)+MID$(TA$(J),NO*3-2,3)
1800  NEXT J
1810  LOCATE 32,2+I:PRINT ":"+NA$(I)
1820  SP(I)=VAL(MID$(TA$(4),NO*3-2,3))
1830  DP(I)=VAL(MID$(TA$(5),NO*3-2,3))
1840  AP(I)=VAL(MID$(TA$(6),NO*3-2,3))
1850  WP(I)=VAL(MID$(TA$(7),NO*3-2,3))
1860 NEXT I
1870 GOTO "GAME"
1880 ' --- END ---
1890 LABEL "END"
1900 CLS:PRINT " ** END **"
1910 PRINT "Are you sure? (Y/N)"
```

```
1920 K$=INKEY$:IF K$="" THEN 1920
1930 IF K$<>"Y" AND K$<>"y" THEN "MENU"
1940 CONSOLE:CLS4:REPEAT ON:END
1950 ' --- FIGHT ---
1960 LABEL "GAME"
1970 CLS4: COLOR 7,1:CONSOLE 0,25,0,40:WIDTH 40:CGEN1:KMODE0:CBLACK(1)
1980 FOR I=12 TO 26 STEP 2
1990  LOCATE I,16:PRINT CH$(0,15)
2000 NEXT I
2010 LOCATE 12,14:PRINT CH$(1,14)
2020 LOCATE 26,14:PRINT CH$(0,14)
2030 SCREEN 0,0
2040 HP(0)=100:HP(1)=100:X(0)=24:X(1)=14
2050 LOCATE X(0),14:PRINT CH$(0,0)
2060 LOCATE X(1),14:PRINT CH$(0,0)
2070 CGEN 0:KMODE1
2080 LOCATE 6,22:PRINT NA$(1)
2090 LOCATE 25,22:PRINT NA$(0)
2100 CGEN 1:KMODE0
2110 LOCATE 8,2:PRINT KU$
2120 LOCATE 25,18:PRINT KU$
2130 LOCATE 1,11:PRINT KU$
2140 LOCATE 29,3 :PRINT KU$
2150 FOR I=0 TO 2
2160 LINE (70+I*200,150)-(50+I*200,126),PSET,2  ,BF
2170 LINE (70+I*200,125)-(50+I*200,101),PSET,6  ,BF
2180 LINE (70+I*200,100)-(50+I*200,50),PSET,5  ,BF
2190 NEXT I
2200 CGEN 0
2210 LOCATE 16,8:PRINT "GO FIGHT!!"
2220 PAUSE 10
2230 LOCATE 15,8:PRINT SPACE$(15)
2240 CGEN 1
2250 ' --- MOVE ---
2260 FOR Z=0 TO 1
2270  W=Z XOR 1
2280  Y=SGN(INT(RND*SP(Z))-SP(Z)/2+.5)
2290  L=Y
2300  IF CHARACTER$(X(Z)+Y+W,14)<>CHR$(&HAC+Z*65) AND X(Z)-X(W)<>-2*Y THENX(Z)=X
(Z)+Y:ELSE L=0
2310  I(Z)=(I(Z)+Y*SGN(Z-.5)) AND 3
2320  IF Y<>0 THEN LOCATE X(Z)-L,14 :PRINT SP$:ELSE LOCATE  X(Z),14  :PRINT CH$(
Z,9):GOTO 2340
2330  LOCATE  X(Z),14  :PRINT CH$(Z,I(Z)+1)
2340  IF X(0)-X(1)<4 AND RND>.3 THEN 2630
2350 NEXT Z
2360 ' --- WEAPON ---
2370 IF X(0)-X(1)<4 THEN 2250
2380 Z=(RND*1)
2390 W=Z XOR 1
2400 V=Z-W
2410 IF INT(RND*SP(W)/WP(Z)*5)<>0 THEN 2260
2420 LOCATE X(Z),14:PRINT CH$(Z,5)
2430 PLAY"O4B5AGFEDC"
2440 LOCATE X(Z)+2*V,14:PRINT CH$(Z,INT(WP(Z)/15)+11)
2450 FOR XX=X(Z)+3*V TO X(W)-2*V STEP V
2460  LOCATE XX-V,14:PRINT SP$
2470  LOCATE XX,14:PRINT CH$(Z,INT(WP(Z)/15)+11)
2480 NEXT XX
2490 LOCATE X(W)-2*V,14:PRINT SP$
2500 IF INT(RND*DP(W)/AP(Z)*5 )=0 THEN LOCATE X(W),10:GOSUB 2770:GOTO 2600
2510 LOCATE X(W),14:PRINT SP$
2520 LOCATE X(W),12:PRINT CH$(W,1)
2530 FOR XX=X(W)-1*V TO 19+5*V STEP V
2540  LOCATE XX-V,14:PRINT SP$
2550  LOCATE XX,14:PRINT CH$(Z,INT(WP(Z)/15)+11)
2560 NEXT XX
2570 LOCATE XX-V,14:PRINT SP$
2580 LOCATE X(W),12:PRINT SP$
2590 LOCATE X(W),14:PRINT CH$(W,9)
2600 LOCATE X(Z),14:PRINT CH$(Z,9)
2610 I(0)=4:I(1)=4
2620 GOTO 2250
2630 ' --- ZUTUKI ---
2640 Z=(RND*1)
2650 W=Z XOR 1
2660 LOCATE X(Z),14:PRINT CH$(Z,6)
2670 PAUSE 2
2680 F=X(Z)+FIX((X(W)-X(Z))/3)
2690 LOCATE X(Z),14:PRINT SP$
2700 LOCATE F,14:PRINT CH$(Z,7)
2710 IF INT(RND*DP(W)/AP(Z)*5)<>0 THEN PLAY "O7A5":GOTO 2730
2720 GOSUB 2770
2730 PAUSE 2
2740 LOCATE F,14:PRINT SP$
2750 LOCATE X(Z),14:PRINT CH$(Z,9)
2760 GOTO 2260
2770 ' --- DAMAGE ---
2780 LOCATE X(W),14:PRINT CH$(W,8)
2790 PLAY"O2A9O5A9"
2800 LOCATE F,14:PRINT SP$
2810 LOCATE X(Z),14:PRINT CH$(Z,9)
2820 IF X(0)-X(1)<4 THEN DM=(RND*AP(Z))+2 ELSE DM=(RND*WP(Z)*2)+2
2830 HP(W)=HP(W)-DM
2840 LINE (50+200*Z,50)-(70+200*Z,150-HP(W)),PRESET,,BF
2850 IF HP(W)<=0 THEN GOTO 2870
2860 RETURN
2870 ' --- GAME OVER ---
2880 PAUSE 3:LOCATE X(W),14:PRINT CH$(W,10)
2890 PAUSE 5:LOCATE X(Z),14:PRINT CH$(0,0)
2900 PAUSE 6:LOCATE X(Z),11:PRINT CH$(1,15)
2910 CGEN 0:KMODE 1
2920 LOCATE 12,5:PRINT "Won by player";Z+1
2930 LOCATE 12,7:PRINT "  :";NA$(Z)
2940 RESTORE 4920:READ MU1$,MU2$,MU3$
2950 TEMPO 200:PLAY MU1$+":"+MU2$+":"+MU3$
2960 CGEN 1:LOCATE X(Z),14:PRINT CH$(1,0)
```

▶最近新作のゲームをやった事がない。それにしてもGAME REVIEWのサイオブレードのゲーム要素での「下着ドロボウ」というのはいったいなんなんだ？

池田 和繁 (18) 大分県

```
2970 CGEN 0:LOCATE 14,9:PRINT "Hit any key."
2980 WHILE INKEY$="":WEND
2990 GOTO "PRINT1"
3000 ' --- ERROR ---
3010 LABEL "ERROR"
3020 IF ERR=60 THEN BEEP:PRINT "Device full":PAUSE 10:RESUME "MENU"
3030 IF ERR=53 THEN RESUME 170
3040 PRINT "ERROR";ERR;" IN";ERL:STOP
3050 ' --- CHARA SET ---
3060 LABEL "PCGSET"
3070 GOSUB 3190
3080 FOR O=0 TO 1
3090  FOR J=0TO 1
3100   FOR I=0 TO 7
3110    A=J*64+I*2+&H80+O*32
3120    CH$(J,I+O*8)=CHR$(A,1+A,&H1F,&H1D,&H1D,16+A,17+A)
3130   NEXT I
3140  NEXT J
3150 NEXT O
3160 SP$=CHR$(&H30,&H30,&H1F,&H1D,&H1D,&H30,&H30)
3170 KU$=CHR$(112,113,114,115,116,31,29,29,29,29,29,117,118,119,120,121,122,31,2
9,29,29,29,29,29,123,124,125,126,127)
3180 RETURN
3190 ' --- PCG ---
3200 DEFCHR$(&H20)=HEXCHR$("0000000000000000000000000000000000000000000000000")
3210 DEFCHR$(&H30)=HEXCHR$("0000000000000000000000000000000000000000000000000")
3220 DEFCHR$(&H31)=HEXCHR$("3C7EFFFFFFFF7E3C00000C0C00000000001C3E3E3E1C0000")
3230 DEFCHR$(112)=HEXCHR$("0001070F1F3F3F7F000001070F1F1F3F000001070F1F1F3F")
3240 DEFCHR$(113)=HEXCHR$("3EFFFFFFFFFFFFFF003EFFFFFFFFFFFF003EFFFFFFFFFFFF")
3250 DEFCHR$(114)=HEXCHR$("00C0F0FCFEFFFFFF0000C0F0FCFEFFFF0000C0F0FCFEFFFF")
3260 DEFCHR$(115)=HEXCHR$("00000000007FFFF00000000000007FF00000000000007FF")
3270 DEFCHR$(116)=HEXCHR$("000000000000C0F000000000000000C0000000000000000C0")
3280 DEFCHR$(117)=HEXCHR$("7F7FFFFFFFFF7F7F3F3F7F7F7F7F3F3F3F3F7F7F7F7F3F3F")
3290 DEFCHR$(118)=HEXCHR$("FFFFFFFFFFFFFFFFF1CEBF7F7F7FB7CFF1CEBF7F7F7FB7CF")
3300 DEFCHR$(119)=HEXCHR$("FFFFFFFFFFFFFFFFFFFFF7FBFBFDFDFDFFFFF7FBFBFDFDFDF")
3310 DEFCHR$(120)=HEXCHR$("FFFFFFFFFFFFFFFFFFFFFFFFF1EEDFDFDBFFFFFFF1EEDFDFDB")
3320 DEFCHR$(121)=HEXCHR$("F8FCFFFFFFFFFFFFF0F8FCFFFF7FBFBEF0F8FCFFFF7FBFBE")
3330 DEFCHR$(122)=HEXCHR$("0000E0FEFFFEF08000000E0FEF0800000000E0FEF08000")
3340 DEFCHR$(123)=HEXCHR$("7F3F3F1F0F0701003F1F1F0F070100003F1F1F0F07010000")
3350 DEFCHR$(124)=HEXCHR$("FFFFFFFFFFFFCF000FFFFFFFFFCF00000FFFFFFFFFCF00000")
3360 DEFCHR$(125)=HEXCHR$("FFFFFFFF70000000BFBFBF7000000000BFBFBF7000000000")
3370 DEFCHR$(126)=HEXCHR$("FFFFFFFCE0000000E7FFFCE000000000E7FFFCE000000000")
3380 DEFCHR$(127)=HEXCHR$("FEFCF8E000000000BC78E00000000000BC78E00000000000")
3390 DEFCHR$(128)=HEXCHR$("030C13274F5F9F9F00030C1B3522616000030C1B372F6F6F")
3400 DEFCHR$(129)=HEXCHR$("C030C8E4F2FAF9F900C030D8ECF476B600C030D8ECF4F6F6")
3410 DEFCHR$(130)=HEXCHR$("00000F101F3F3F7F0000000F00070B050000000F001F1F3F")
3420 DEFCHR$(131)=HEXCHR$("0000C0300884C4E2000000C0F078B8DC000000C0F078B8DC")
3430 DEFCHR$(132)=HEXCHR$("07080E1F3F7F7F7F0007010E170B05020007010E1F3F3F3F")
3440 DEFCHR$(133)=HEXCHR$("C030080484C2C2C200C0F0F878BCBCBC00C0F0F878BCBCBC")
3450 DEFCHR$(134)=HEXCHR$("000007080C1E3F7F00000007030D0E1600000007030D1E3E")
3460 DEFCHR$(135)=HEXCHR$("0000C030080404020000000C0F0F8F8FC000000C0F0F8F8FC")
3470 DEFCHR$(136)=HEXCHR$("07080E1F3F7F7F7F0007010E170B05020007010E1F3F3F3F")
3480 DEFCHR$(137)=HEXCHR$("C030080484C2C2C200C0F0F878BCBCBC00C0F0F878BCBCBC")
3490 DEFCHR$(138)=HEXCHR$("071F3F7F7FBE5C2000070B050241231F00071F3F3E5D231F")
3500 DEFCHR$(139)=HEXCHR$("C0B08884020202020040707 8FCFCFCFC00407078FCFCFCFC")
3510 DEFCHR$(140)=HEXCHR$("01070F1F3F3F2F27000102010000101800001070F1F0F1718")
3520 DEFCHR$(141)=HEXCHR$("C0F8E4E2E1C1810100C0D85C9E3E7EFE00C0D8DCDEBE7EFE")
3530 DEFCHR$(142)=HEXCHR$("07182040408 0B8FC40871F3F3F7F471B40871F3F3F7F473B")
3540 DEFCHR$(143)=HEXCHR$("80601008040404040080E0F0F8F8F8F80080E0F0F8F8F8F8")
3550 DEFCHR$(144)=HEXCHR$("8F874020100B000070783F1F0F041E3E77783F1F0B000000")
3560 DEFCHR$(145)=HEXCHR$("F1E1020408D000004E1EFCF8F020787CEE1EFCF8D0000000")
3570 DEFCHR$(146)=HEXCHR$("7F7F7F7FC0E0F87702010000BFDFE7703F3F3F003F1F0700")
3580 DEFCHR$(147)=HEXCHR$("E2E2C282020018E0DC5CBC7CFDFFE71EDCDCBC7CFCF8E000")
3590 DEFCHR$(148)=HEXCHR$("7F7F402018070F1F01003F1F07000F1F3F003E1E06000000")
3600 DEFCHR$(149)=HEXCHR$("82020204080080807CFCFCF8F0F0F0807CFC3C1800000000")
3610 DEFCHR$(150)=HEXCHR$("7F7F7E7C402018000A0401033F1F07033E3E3D033B190000")
3620 DEFCHR$(151)=HEXCHR$("0202020202041C38FCFCFCFCFCF8E4D8FCFCFCFCFCF86000")
3630 DEFCHR$(152)=HEXCHR$("7F7F4060703F1C0001003F5F6F301C013F003F1F0F000000")
3640 DEFCHR$(153)=HEXCHR$("8202020418C000007CFCFCF8E020F8F87CFCFCF8E0000000")
3650 DEFCHR$(154)=HEXCHR$("1C030430403F0E1C03301B0F3F000F1F0300030F3F000000")
3660 DEFCHR$(155)=HEXCHR$("0202040810A00000FCFCF8F0E040E0E0FCFCF8F0E0000000")
3670 DEFCHR$(156)=HEXCHR$("2020100804031C381F1F0F0703001F3F1F1F0F0703000000")
3680 DEFCHR$(157)=HEXCHR$("0101020418A00000FEFEFCF8E040E0E0FEFEFCF8E0000000")
3690 DEFCHR$(158)=HEXCHR$("FE7F7F3F1E0C03012D160A04814300017D3E3E1E8D430000")
3700 DEFCHR$(159)=HEXCHR$("0404081020F0E0C0F8F8F0E0D63EFCF8F8F8F0E0C0000000")
3710 DEFCHR$(160)=HEXCHR$("00000000000000103012111090 4C03001012111090 4C03001")
3720 DEFCHR$(161)=HEXCHR$("0000304874F4F4F20000000300868E86C000000300868E8EC")
3730 DEFCHR$(162)=HEXCHR$("07080E1F3F7F7F7F0007010E170B05020007010E1F3F3F3F")
3740 DEFCHR$(163)=HEXCHR$("C030080484C2C2C200C0F0F878BCBCBC00C0F0F878BCBCBC")
3750 DEFCHR$(164)=HEXCHR$("0000070910002020001C3876EFDF1F1F000000060F0F1F1F")
3760 DEFCHR$(165)=HEXCHR$("0000E0FCFAFDFDFD000000E0743A5A2A000000E0747A7A7A")
3770 DEFCHR$(166)=HEXCHR$("0000000000003E7F0000000000000001C000000000000003E7F")
3780 DEFCHR$(167)=HEXCHR$("000000000000008000000000000000000000000000000000080")
3790 DEFCHR$(168)=HEXCHR$("00000000F3F7FFF00000000000001F3C00000000F3F7FFF")
3800 DEFCHR$(169)=HEXCHR$("00000000E080F8E000000000000000800000000000E080F8E0")
3810 DEFCHR$(170)=HEXCHR$("0000071F3F7FFFFF000000000030F1F3F0000071F3F7FFFFF")
3820 DEFCHR$(171)=HEXCHR$("00FEF8E0FEF8FFF80000004080E000C000FEF8E0FEF8FFF8")
3830 DEFCHR$(172)=HEXCHR$("1F20408C98A9AB4400000000000000000001F3F7367464400")
3840 DEFCHR$(173)=HEXCHR$("008C5221060101C9000000000E0F03000008CDEF8DECE06")
3850 DEFCHR$(174)=HEXCHR$("000004044AD5AA5500000002050A55AA7FFFFFFDF8B00000")
3860 DEFCHR$(175)=HEXCHR$("0000205062D2AE550000000201028 51AAFEFFDFCFCF870000")
3870 DEFCHR$(176)=HEXCHR$("0F1F3F3FDFE3F07F00000000A0DCEF70030F1F1F231C0F00")
3880 DEFCHR$(177)=HEXCHR$("F2F2F2E2C28000E0AC4C2C1C3D7FFF1EECECECDCBC78E000")
3890 DEFCHR$(178)=HEXCHR$("7F7F402018070E1C01003F1F07000F1F3F003F1F07000000")
3900 DEFCHR$(179)=HEXCHR$("8202020418A000007CFCFCF8E040F0F07CFCFCF8E0000000")
3910 DEFCHR$(180)=HEXCHR$("2021010000008071F1E7EFFFF7F07001F1E060303070700")
3920 DEFCHR$(181)=HEXCHR$("FDF9F1E2020418E012060E1CFCF8E0007AF6EE1CFCF8E000")
3930 DEFCHR$(182)=HEXCHR$("FFFFFF7F3E0000003E3F3E1C00000000FFFFFF7F3E000000")
3940 DEFCHR$(183)=HEXCHR$("F8FFF8800000000000F000000000000000F8FFF8800000000")
3950 DEFCHR$(184)=HEXCHR$("FFFFFFFF7F3F0F003F3E3F3C1F000000FFFFFFFF7F3F0F00")
3960 DEFCHR$(185)=HEXCHR$("FEF0FFE0F8C0E000C000E000C0000000FEF0FFE0F8C0E000")
3970 DEFCHR$(186)=HEXCHR$("FFFFFF7F3F1F07003E3F1F0F01000000EFFFFF7F3F1F0700")
3980 DEFCHR$(187)=HEXCHR$("FFE0FEF0FCF0FCFF00C00080C0600000FFE0FEF0FCF0FCFF")
3990 DEFCHR$(188)=HEXCHR$("0000000000000001000000000000000000000000000000000")
4000 DEFCHR$(189)=HEXCHR$("2526242424448402181818181183878FC0200000000000000")
4010 DEFCHR$(190)=HEXCHR$("AA552A150201000055AA552A050201000000000000000000")
4020 DEFCHR$(191)=HEXCHR$("AA55AA5480000000055AA54AA4480000000000000000000000")
```

▶北海道の今年の冬は，寒い日，暖かい日，雪が降る日，降らない日が，まるでサイコロでも振って決められているようだ。今日なんか雪というよりブリザードみたいで視界ゼロ。雪のないところで1年ほど過ごしてみたい。雪国以外の人にこの気持ちわかるでしょうか。雪はロマンチックなんかじゃないぞー！　　中鉢　進一 (20) 北海道

```
4030 DEFCHR$(192)=HEXCHR$("0F172F5F5F9F8F80000912212060707F000B172F2F6F707F")
4040 DEFCHR$(193)=HEXCHR$("F0E8F4FAFBF9F10100D0E874B4560EFE00D0E8F4F4F60EFE")
4050 DEFCHR$(194)=HEXCHR$("0000030C102123470000000030F1E1C3800000000030F1E1D3B")
4060 DEFCHR$(195)=HEXCHR$("0000F008F8FCFCFE000000F00078B85C000000F000F8F8FC")
4070 DEFCHR$(196)=HEXCHR$("030C102021434343000030F1F1E3C3C3C00030F1F1E3D3D3D")
4080 DEFCHR$(197)=HEXCHR$("E01070F8FCFEFEFE00E08070B85C2C1400E08070F8FCFCFC")
4090 DEFCHR$(198)=HEXCHR$("0000030C10202040000000030F1F1F3F000000030F1F1F3F")
4100 DEFCHR$(199)=HEXCHR$("0000E0103078FCFE000000E0C0B0385C000000E0C0B0787C")
4110 DEFCHR$(200)=HEXCHR$("030C1020214343430030F1F1E3C3C3C00030F1F1E3D3D3D")
4120 DEFCHR$(201)=HEXCHR$("E01070F8FCFEFEFE00E08070B85C2C1400E08070F8FCFCFC")
4130 DEFCHR$(202)=HEXCHR$("030D11214040404000020E1E3F3F3F3F00020E1E3F3F3F3F")
4140 DEFCHR$(203)=HEXCHR$("E0F8FCFEFE7D3A040060B85C2C92C4F800E0F8FC7CBAC4F8")
4150 DEFCHR$(204)=HEXCHR$("031F274787838180000011A39787C7E7F00031B3B7B7D7E7F")
4160 DEFCHR$(205)=HEXCHR$("80E0F0F8FCFCF4E40080E070B8502818080E0F0F8F0E818")
4170 DEFCHR$(206)=HEXCHR$("010608102020202000001070F1F1F1F1F0001070F1F1F1F1F")
4180 DEFCHR$(207)=HEXCHR$("E01804020211D3F02E1F8FCFCFEE2DC02E1F8FCFCFEE2DC")
4190 DEFCHR$(208)=HEXCHR$("9F9F4E24100B0000606031 1B0F041E3E606030180F000000")
4200 DEFCHR$(209)=HEXCHR$("F9F9722408D0000006068CD8F020787C06060C18F0000000")
4210 DEFCHR$(210)=HEXCHR$("47474341400018073838 3C3EBFFFE7783B3B3D3E3F1F0700")
4220 DEFCHR$(211)=HEXCHR$("FEFEFEFE03071FEE2C140800FDFBE70EFCFCFC00FCF8E000")
4230 DEFCHR$(212)=HEXCHR$("414040201000010 13E3F3F1F0F0F0F013E3F3C1800000000")
4240 DEFCHR$(213)=HEXCHR$("FEFE020418E0F0F80800FCF8E000F0F8FC007C7860000000")
4250 DEFCHR$(214)=HEXCHR$("404040404020381C3F3F3F3F3F1F271B3F3F3F3F3F1F0600")
4260 DEFCHR$(215)=HEXCHR$("FEFE7E3E020418002C1488C0FCF8E0C07C7CBCC0DC980000")
4270 DEFCHR$(216)=HEXCHR$("41404020180300003E3F3F1F07041F1F3E3F3F1F07000000")
4280 DEFCHR$(217)=HEXCHR$("FEFE02060EFC38000800FCFAF60C3880FC00FCF8F0000000")
4290 DEFCHR$(218)=HEXCHR$("40402010080500003F3F1F0F070207073F3F1F0F07000000")
4300 DEFCHR$(219)=HEXCHR$("38C0200C02FC7038C00CD8F0FC00F0F8C000C0F0FC000000")
4310 DEFCHR$(220)=HEXCHR$("80804020180500007F7F3F1F070207077F7F3F1F07000000")
4320 DEFCHR$(221)=HEXCHR$("0404081020C0381CF8F8F0E0C000F8FCF8F8F0E0C0000000")
4330 DEFCHR$(222)=HEXCHR$("20201008040F07031F1F0F076B7C3F1F1F1F0F0703000000")
4340 DEFCHR$(223)=HEXCHR$("7FFEFEFC7830C0809E2C140881C20080BE7C7C78B1C20000")
4350 DEFCHR$(224)=HEXCHR$("00000C122E2F2F4F0000000C1016133500000000C10161737")
4360 DEFCHR$(225)=HEXCHR$("000000000000080C080848890200 30C808084889020030C80")
4370 DEFCHR$(226)=HEXCHR$("030C1020214343430030F1F1E3C3C3C00030F1F1E3D3D3D")
4380 DEFCHR$(227)=HEXCHR$("E01070F8FCFEFEFE00E08070B85C2C1400E08070F8FCFCFC")
4390 DEFCHR$(228)=HEXCHR$("0000073F5FBFBFBF000000072E564A4400000000072E5E5E5E")
4400 DEFCHR$(229)=HEXCHR$("0000E09008000404 00381C6EF7FBF8F800000060F0F0F8F8")
4410 DEFCHR$(230)=HEXCHR$("000000000000000100000000000000000000000000000001")
4420 DEFCHR$(231)=HEXCHR$("0000000000007CFE000000000000000038000000000000007CFE")
4430 DEFCHR$(232)=HEXCHR$("000000007011F07000000000000000100000000000007011F07")
4440 DEFCHR$(233)=HEXCHR$("0000000F0FCFEFF0000000000000F83C00000000F0FCFEFF")
4450 DEFCHR$(234)=HEXCHR$("007F1F077F1FFF1F000000020107000 3007F1F077F1FFF1F")
4460 DEFCHR$(235)=HEXCHR$("0000E0F8FCFEFFFF00000000C0F0F8FC0000E0F8FCFEFFFF")
4470 DEFCHR$(236)=HEXCHR$("00314A8460808093000000000000070F0C0000317B1F7B7360")
4480 DEFCHR$(237)=HEXCHR$("F80402311995D52200000000000000000000F8FCCEE6622200")
4490 DEFCHR$(238)=HEXCHR$("1440008000801F1B0A3F7E7CFA70FF5F146A54A852A05F1B")
4500 DEFCHR$(239)=HEXCHR$("280200010001F8F850FC7E3E5F0EFFFA28562A154A05FAF8")
4510 DEFCHR$(240)=HEXCHR$("4F4F4F474301000732313038BCFEFF783737373B3D1E0700")
4520 DEFCHR$(241)=HEXCHR$("F0F8FCFCFBC70FFEC070B858053BF70EC0F0F8F8C438F000")
4530 DEFCHR$(242)=HEXCHR$("41404020180500003E3F3F1F07020F0F3E3F3F1F07000000")
4540 DEFCHR$(243)=HEXCHR$("FEFE020418E070380800FCF8E000F0F8FC00FCF8E0000000")
4550 DEFCHR$(244)=HEXCHR$("BF9F8F4740201807426070383F1F07005E6F77383F1F0700")
4560 DEFCHR$(245)=HEXCHR$("0484800000010E0F8787EFFFFFEE000F87860C0C0E0E000")
4570 DEFCHR$(246)=HEXCHR$("1FFF1F010000000000F000000000001FFF1F0100000000")
4580 DEFCHR$(247)=HEXCHR$("FFFFFFFE7C0000007CFC7C3800000000FFFFFFFE7C000000")
4590 DEFCHR$(248)=HEXCHR$("7F0FFF071F030700030007000300000007F0FFF071F030700")
4600 DEFCHR$(249)=HEXCHR$("FFFFFFFFFEFCF000FC7CFC3CF8000000FFFFFFFFFEFCF000")
4610 DEFCHR$(250)=HEXCHR$("FF077F0F3F0F3FFF00030001030600000FF077F0F3F0F3FFF")
4620 DEFCHR$(251)=HEXCHR$("FFFFFFFEFCF8E0007CFCF8F080000000FFFFFFFEFCF8E000")
4630 DEFCHR$(252)=HEXCHR$("A4642424242221401818181818 1C1E3F4000000000000000")
4640 DEFCHR$(253)=HEXCHR$("00000000000080000000000000000000000000000000000000")
4650 DEFCHR$(254)=HEXCHR$("911D5B919B3A1F1F9F1F5F1F9F1F5F1F111D1B911B3A5F1F")
4660 DEFCHR$(255)=HEXCHR$("08680A5858C9F8F8F9F8F8F8FCF9F8F809680A585CC8F8F8")
4670 RETURN
4680 ' --- TITLE DATA ---
4690 DATA 8,8,8,8,8,8,8,127
4700 DATA 65,65,127,65,65,34,20,8
4710 DATA 65,65,65,65,73,85,99,65
4720 DATA 65,65,127,65,65,34,20,8
4730  ' --- TAMA OPENING MUSIC DATA (by Kouji Mori) ---
4740 DATA O3C5G5o2G5o3G3C6G5o2G5o3G5C5G5o2G5o3G3C6G5o2G5o3G5,R1R1,R1R1
4750 DATA O3C5G5o2G5o3G3C6G5o2G5o3G5C5G5o2G5o3G3C6G5o2G5o3G5
4760 DATA O4G9G7E5GA6G6#F5G7#F7,o4E9E7C5EF6E6#D5E7#D7
4770 DATA O3C5G5o2G5o3G3C6G5o2G5o3G5C5G5o2G5o3G3C6G5o2G5o3G5
4780 DATA F9F7D5F5G6F6G5E9,D7D9o3B5o4D5E6D6E5C9
4790 DATA O3C5G5o2G5o3G3C6G5o2G5o3G5C5G5o2G5o3G3C6G5o2G5o3G5
4800 DATA O4G9G7E5GA6G6#F5G7#F7,o4E9E7C5EF6E6#D5E7#D7
4810 DATA O3D5G5o2G5o3G3D6G5o2G5o3G5C5G5o2G5o3G3C5C3G5C5D3E
4820 DATA F9F6G5G3F5E6F6E5E9,D9D6E5E3D5C6D6C5C9
4830 DATA O3F3FAACCAF5F3AACCAA,A6B6A6A5A3BBA5,F6G6F6F5F3GGF5
4840 DATA E3EED5D3CCFFAACCAA,o5C6o4G6E5A7G7,E6D6C5F7E7
4850 DATA D3DGGo2GGo3GD5D3GGo2GGo3GG,D6E6D6F6E5D5,F6G6F6A6G5F5
4860 DATA C3CCo2G5GG3o3CCDEFGAB,C6D6C5C9,E6F6E5E9
4870 DATA O3F3FAACCAF5F3AACCAA,A6B6A6A5A3BBA5,F6G6F6F5F3GGF5
4880 DATA E3EED5D3CCFFAACCA5,o5C6o4Bo5C5o4A7G,E6D6E5C7o3B7
4890 DATA D5#F3#Fo2AAo3#FD5D3#F#Fo2AAo3#F5,#F6D#F#FD5#F5,o4D6o3A6o4D6D6C3CCC
4900 DATA G5G3D5DD3G5RBR,G6AG5BRGR,o3B6BB5G5RG3GAB
4910 DATA 0,0,0
4920 DATA O2A5A3F5F3AAB5B3G5G3AB+C5+C3-G5G3EEC7+C
4930 DATA O4A6FA5B6G6B5+C6-G6E5C7+C5
4940 DATA O3F6F6F5G6G6G5E6E6E5E
4950 '
4960 '       T.A.M.A.          FOR X1/turbo
4970 '
4980 '  でざいんの係
4990 '  きゃらくたーの係
5000 '  ぷろぐらむの係          :岩崎直明      in FLOPS
5010 '  おんがくの係            :森浩二        in FLOPS
5020 '  あそぶ係                :ふらっぺ竹丸  in FLOPS
5030 '                          :れいとれ富樫
5040 '                          :えすぱー沼田
5050 '                          :ろーりー西村
5060 '                          :そのたの皆様
```

C言語の概要を見る

Sohma Hidetomo
相馬 英智

OS-9での開発の主力はC言語となります。今回は，新しいOSの主要開発ツールをレポートしてみましょう。まだまだアプリケーションは揃っていませんが，C言語によってユーザーレベルでの環境整備も可能になります。

ここ数カ月にわたってOS-9/68000についての説明がなされてきました。そこで今回は回り回って，OS-9/X68000用のCについて見ていきたいと思います。今回リリースされる「C＆プロフェッショナルパッケージ」にはC言語以外にもさまざまなツール類が付属していますが，ここではC言語にしぼってお話しします。まずはCの概要を中心とし，詳細な内容については製品版が発売されてからのお楽しみとしておきましょう。

これまでに一般のOS-9/68000用としてリリースされているCはMicroware C V2.2ですが，X68000版としてリリースされるCはMicroware C V3.1相当のものでさらにX68000専用の拡張がなされています。OS-9/X68000ではCがプログラム開発の主流言語として扱われているようです。

UNIXとの関係

まず初めにいっておきたいのは，このOS-9/X68000のCがUNIXのC（バージョン7）をかなり意識したものであるということです。というよりは，このマイクロウェア社のOS-9/68000自身がきわめてUNIXを意識しているといえます。

さて，ここでUNIXってなんだろうと思っておられる方のためにちょっと説明しますと，UNIXは現在ワークステーションクラス以上のコンピュータでよく使われているOS（オペレーティングシステム）でマルチプロセス（いくつもの仕事が並行してできる）とマルチユーザー（複数の人が同時に使える）といった特徴を持っています。

また，大学などの教育機関や研究機関では昔から比較的手に入りやすかったので（ほとんど“ただ”だった），その方面でよく使われています。しかし，最近はいろいろなところで使われていますので，あまり限定はできません。独自の全国的ネットワークで結ばれており，開発されるプログラムの多くがパブリックドメイン（著作権のないプログラム）として公開されているため，たくさんの優秀なアプリケーションが時代の最先端をいく技術者の手によって絶えず改良/開発されています。

このOS-9/X68000とそのCは，このUNIXを強く意識したものとなっています。つまりUNIXの長所をできるだけ取り込もうとしており，UNIXと相性がよくなるように作られているのです。ということで，ここでのCの説明はUNIXのCとHuman68kのXCと，このOS-9/X68000のCの比較のようなかたちで行いたいと思います。

概要を見る

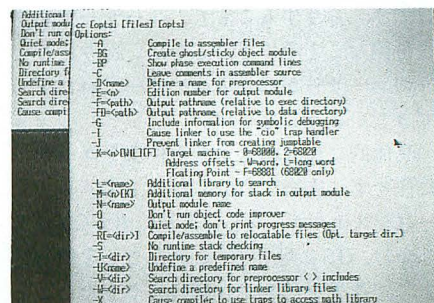
まずこのCの全体の感じはといいますと，いわゆるC言語の標準的な仕様と思われるものにはほとんど対応しており，これといって妙な方言があったり，使いにくかったりといった問題はありません。

さて，ここでいうCの標準的な仕様とは，かの有名な“The C Programing Language（日本語訳『プログラミング言語C』通称K＆R）”という本に書かれているものを指しています。もともと，この本でC言語自体が定義されているようなもので，すべてのC言語処理系はこの本を無視するわけにはいかないのです。

ここでK＆Rに則しているということで，Cの入門書に書いてあるようなプログラムはほとんどの場合変更などの必要はなく，そのまま動くことが期待できます。ただ，細かな仕様についてはXCなどと若干の違いが見られます。

ライブラリ関数について

Cは関数指向の言語なので，どのような関数がサポートされているのかということはかなり重要な問題です。このCは以下に述べるような，大きく分けて4種類の関数をサポートしています。

まず，1種類目はごく標準的なCの関数群です。これらの関数はCにとってのきわめて基本的な動作を行うもので入出力など

コンパイルオプションもたくさん

を行うものを中心にしたもので，いままでのCの歴史のなかでも最初からほとんど使い方が変更されなかったものです。

具体的には，インクルードファイル“stdio.h”で定義される標準入出力関数，一般関数，文字および文字列に対して処理を行う関数，そして算術演算を行う関数です。これらについてはほかのCについても同様に，標準的なCであるといわれているCのあいだでは，ほとんど同じように使えるようになっています。

逆に，このあたりから使い方が異なるようでは，方言の激しいCといわれてもしかたがないでしょう。ただし，XCはこれらの関数についても若干の拡張をしています（具体的には，文字列処理関数など）。

したがって，もしそのような関数を使ったプログラムをこのOS-9/X68000で走らせたい場合は若干の修正が必要です。

しかし，これは比較的簡単な修正ですむものと思われます。というのは，このような修正はほかの関数の簡単な組み合わせや，マクロ定義などで比較的単純に対処できるからです。移植をあまり頻繁に行うようであれば，XCからの移植用のインクルードファイルを作ってしまうのがよいでしょう。そうすれば，以後は面倒な作業から解放されます。

システムコール関数

関数の2種類目は，OS-9のシステムコールを行うための関数群です。さて，システムコールとはなんだと思われる方もおいで

でしょう。システムコールとは，ユーザーのプログラムがOSに対してOSの持っている機能を要求し，その作業をOSにやってもらうことをいいます。UNIXやOS-9などのマルチプロセスのOSになると，ユーザーにあまり勝手なことをされると，OSが止められたり破壊されたりして問題が起きる可能性があります。

そんなことが起きると困るので，OS自身が危険性を伴うものに対して“仕切る”わけです。そうすることで，ユーザーが勝手に危険なことをすることを防いだり，ひとりのユーザーがほかのユーザーにいたずらできないようにします。

具体的には，OSは重要な作業をユーザーに任せず，完全に自分で行います。ユーザーはシステムコールというかたちでOSに働きかけ，OSに対して“こんなことをやってよ”と意思を伝えて，自分のやりたいことをやってもらうことになります。

実際のシステムコールにはどのようなものがあるかというと，ファイルの扱いや走っているプロセスについての操作，入出力などです。実は先ほど説明した標準関数などは，その関数を実行する際にいくつかのシステムコールを行って機能の実現を図っています。しかし，このシステムコール関数はそれを直接に行うもので，標準関数などに比べて低水準の関数であるといえます。

さて，システムコールを行う関数はXCにもあるのですが，Human68k のシステムコールとOS-9/X68000のシステムコールではサポートしている機能が違いますので，システムコールの関数を使っているプログラムの移植などを行おうとする場合，結構面倒臭い作業が必要であると思われます（これはMS-DOS用のプログラムに対しても同様）。

UNIXシステムコール関数

関数の3種類目はUNIXのシステムコール関数群です。これは厳密には，UNIXのシステムコールと同じような機能を持っている同名の関数というのが正しいと思われます。UNIXのシステムコール関数をシミュレートする関数というのがもっとも正しいでしょう（リスト1のgetimeなど）。

このような関数群は，明らかにUNIXとOS-9/X68000間のCのプログラムの移植を楽にしようとするためのものです。実際，これによってUNIXのプログラムの移植がかなり容易に行えるようになるでしょう。というのはCにおいてよく使う関数といえば，いちばんが標準関数などで，次がシステムコール関数，残りはあまり使われないからです。

しかし，これによってどのくらい移植しやすくなるかということについては，実際にいくつかのプログラムの移植を行ってみないと詳しいことはいえません。少なくともほかのC（無論UNIX以外のだよ）に比べれば，この関数がある分だけ移植性は向上しているといえるでしょう。

termcap関数その他

最後の関数がtermcap関数と呼ばれるもので，ターミナル（端末）操作を行うための関数です。これは端末の画面の大きさなどの情報を取り出すための関数で端末間の差異に対応できるようなプログラムを書くことを可能にします。

OS-9やUNIXなどのマルチユーザーのOSでは，いろいろ端末から使われることになります。しかし，端末によって使い方が異なったり，動かないプログラムがあったりするときわめて不便です。そこでOSがそれぞれの端末に合わせて動作し，端末間の差異を吸収する必要が生じます。

OS-9やUNIXではどうしているかというと，termcapというファイルに端末の状況を書き込んでおき，OSはこれをもとに端末と交信をして，操作しています。このtermcap関数は現在使われている（プログラムを走らせている）端末の情報をtermcapファイル（termcapデータベース）から取り出すための関数なのです。これから得られたデータをもとに動作を変更するようにプログラムを作ることで，端末に依存しないようなプログラムや，逆に端末の情報を利用して複雑な処理ができるようなプログラムを書くことができるようになります。

このtermcap関数は，単にひとつの端末で使っている場合には関係ないような気がしますが，ウィンドウシステムなどが動き始めますとひとつのウィンドウというのはひとつの端末のようなものですから，プログラムが各ウィンドウによって動作しなかったりするのは困ります。そこで，このような関数があることは，きわめて便利であるといえましょう。

この手の関数はUNIXなどでは標準的なものですが，基本的にひとつの端末からしか使わないHuman68k上では不要のもので，当然のことながらXCではサポートされていません。

ということで，いままで述べたような比較的基本的な関数については特にいうことはありません。あと，このOS-9/X68000のCはウィンドウシステムの操作，日本語への対応やX68000の優れたハードウェア特性を生かせるように，拡張が加えられています。これらはX68000のハードウェアを使用するためのもので，トラップハンドラPSLを使ってプレゼンテーションサポートシステム（PSS）を呼び出します。PSSではさらに高機能な処理が提供されます。

ただ，このような拡張はプログラムの移植の際に問題になる可能性を秘めています。というのは，XCでの拡張とOS-9/X68000用のCの拡張では，拡張のしかたが異なったり，完全に同じものではなかったり，同じ機能の拡張でも関数の名前が違っているようなことが起きる可能性が十分に考えられるからです。これらの場合，ケースバイケースで対応していくしかないものと思われます。

リスト1　UNIXのシステムコール関数を使う

```
==================  moon.c  ==================
     1: /*
     2:     Sample Program.
     3:     The Moon's Age
     4: */
     5:
     6: #include <stdio.h>
     7: #include <time.h>
     8:
     9: main()
    10: {
    11:     struct sgtbuf buffer,*timebuf =& buffer;
    12:     int rev,a;
    13:     float age;
    14:
    15:     getime(timebuf); /* Unix System-Call */
    16:     switch(timebuf->t_month) {
    17:       case 1: rev=1;break;
    18:       case 2: rev=2;break;
    19:       case 3: case 5:rev=-1;break;
    20:       default: rev=0;break;
    21:     }
    22:     age=((timebuf->t_year+160.0)*210.0/19.0-2.0+timebuf->t_month+timebuf-
>t_day+rev)/30.0;
    23:     while (age>1.0) age-=1.0;
    24:     age*=30.0;
    25:     printf("MOON'S AGE = %3.1f¥n",age);
    26: }
```

UNIXのシステムコールと同様の機能が関数として用意されている

データ型について

Cでは基本的な変数のデータ型については結構まとまっているのですが，各データ型がどのように実現されるかについては，機種などに依存するということになっています。このOS-9/X68000用のCでは，データ型は従来のCにANSI拡張の型を加えた，最近では標準的な仕様となっています。

表1にこれらのデータ型のサイズと内部表現を示します。この表によると，int型は32ビット，単精度の浮動小数点型のfloat型は32ビット，倍精度の浮動小数点型のdouble型は64ビットと68000などの68系のCとしては標準的なサイズとなっています。これはXCやUNIXのCの大部分のものと，同じサイズです。このことは，これらのC間でのプログラムの移植性を向上させるであろうと期待できます。

データ型のサイズは意外と注目されていないようですが，プログラムの移植の際にこれでつまずくことが結構あります。

さて，このデータ型のサイズと同様に重要なものに，浮動小数点型の内部フォーマットがあります。この内部フォーマットは，浮動小数点型で計算される値の精度と表現範囲を決定する重要な要素です。図1に，このCで使われている浮動小数点型変数の内部フォーマットを示します。

このフォーマットは，浮動小数点型の内部フォーマットとして標準となりつつあるIEEE方式(IEEE Draft Standard 754)のものに，OS-9/68000標準仕様に合うように一部修正を施したものとなっています。これは浮動小数点コプロセッサMC68881への対応を考慮してのものとなっています。

最後に，記憶クラスについてはどのようになっているかについて述べたいと思います。Cの記憶クラスには，auto, static, extern, registerの4つがありますが，これらについてはちゃんとサポートされています。それどころか，remoteという大きな配列をとることを容易にする記憶クラスが準備され，拡張されています。

ANSI Cに対する拡張

C言語といってもいろいろなCが実際には存在しています。そこで，この事態に対してANSIが乗り出して統一化を図ろうとしています。その統一化の目標とされるCはANSI Cなどと呼ばれ，最近のCはこれの拡張された機能をサポートし，これに基づいたプログラムにも対応しようとしています。これをANSI拡張と呼んでいます。

このANSI拡張は，Cのいろいろな機能の拡張を含んでいるのですが，特に有名なものを挙げるとすると，以下のようなものが挙げられるでしょう。

1) void型，enum型といったデータ型のサポート

void型は値を持たないという型です。これは一見役に立ちそうにないと思えますが，戻り値が必要のない関数の型として使われるものです。このような型で関数の戻り値が不要なことを明示的に示すことによって，この戻り値に対する処理やそれを格納するためのメモリを節約することができます。したがって，その分だけの処理速度の向上と記憶領域の有効活用が期待できます。

また，列挙型(enum型)は全順序のつけられるデータについて，それらを通常の値のようなものとして使えるようにするものです。たとえば1週間の曜日の名前などはどれかの曜日を始まりにすることで，全順序が決まりますので，曜日の名前を配列の添え字などに使えるようになります。

これは，処理速度の向上などにはあまり貢献しませんが，プログラムを読みやすくしてくれます。

2) 構造体同士の値の受け渡し（構造体の関数からの戻り値を含む）

いままでCでは構造体同士の値の受け渡しはできませんでした。それでは，どうしていたかといいますと，構造体を形成する要素（ふつうの変数のようなもの）ごとにデータを移したり，構造体をポインタで押さえておいて参照したりしていました。特に，関数の戻り値を構造体にしたい場合などは，あらかじめ構造体を宣言しておき，その構造体をポインタで参照して値を格納するという，後者の方法がとられていました。

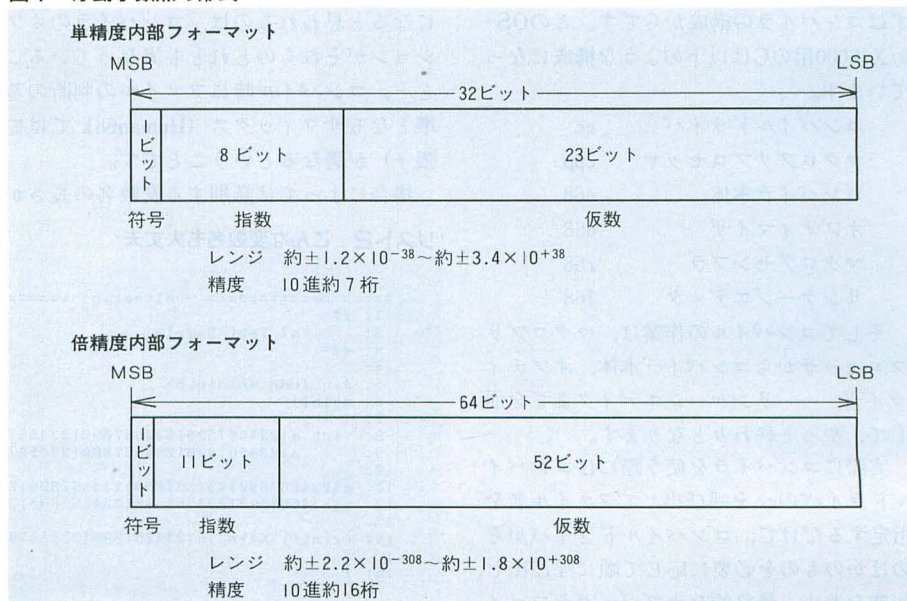
しかし，構造体同士の値の受け渡しが直接行えるようになれば，このような作業はすべて不要となり，構造体がふつうの変数のようなつもりで使えるようになります。はっきりいって，これはたいへん便利です。しかし，構造体は格納領域として使用する記憶領域が大きいものなので，ある構造体に別の構造体の中身のデータを全部コピーするという作業には結構時間がかかってしまいます。これはしかたがないことでしょう。

3) 関数を呼ぶ際の引数のチェック

C言語のプログラムをコンパイルした場合，得られるオブジェクトはほかのふつうのコンパイラ言語(FortranとかPascal など）のものと異なり，プログラムに間違い

表1　OS-9用Cのデータ型

データ型	サイズ(ビット)	内部表現
char	8	2の補数表示
unsigned char	8	符号なし絶対値表示
short	16	2の補数表示
unsigned short	16	符号なし絶対値表示
long	32	2の補数表示
int	32	2の補数表示
unsigned	32	符号なし絶対値表示
float	32	単精度浮動小数点表示
double	64	倍精度浮動小数点表示

図1　浮動小数点の形式

があると即，暴走する可能性があります。これは，Cコンパイラが高速なオブジェクトプログラムを吐き出すために，余計なこと（要するに異常が起きたときなどに対応できるようなもの）をいっさい行わないためです。また，昔のCではエラーのみつけ方が比較的(かなり)甘く，このこともプログラムの暴走の原因になっていました。

そこで，少しでもプログラム内のおかしな部分を発見するために，コンパイルするときに関数を呼び出すときの引数の数と型が関数の中で宣言されているものと合っているかどうかを，調べるようにしようというものです。これで少しは，プログラムの暴走が抑えられ，デバッギングもやりやすくなるものと思われます。

ほかにもいくらかの拡張機能があるのですが，最近のCではとりあえず先の3つぐらいをサポートするのが流行となっているようで，事実XCは先の3つのサポートを行っています。

しかし，それは現時点での話で，そのうち完全にANSI Cと同じ動作ができるCが現れるものと思われます。さて，このOS-9/X68000用のCでもしっかりとANSI拡張が行われています。ただしあくまでもANSI準拠ということで，ANSI拡張のすべてを完全にサポートしているというわけではありません。具体的にはプロトタイプ宣言ができないなどですが，どうしても必要だという人は少ないでしょう。

実際に使ってみると

では使った感じはどうかということを見ていきたいと思います。ということで，まずはコンパイラの構成からです。このOS-9/X68000用のCは以下のような構成になっています。

コンパイルドライバ	cc
マクロプリプロセッサ	cpp
コンパイラ本体	c68
オプティマイザ	o68
マクロアセンブラ	r68
リンケージエディタ	l68

そしてコンパイルの作業は，マクロプリプロセッサからコンパイラ本体，オプティマイザ，……リンケージエディタまでを通して，やっと終わりとなります。

実際にコンパイラを使う際にはコンパイルドライバのccを呼び出してファイル名を指定するだけで，コンパイルドライバがそのほかのものを必要に応じて順に呼び出し，作業を進め，最終的なオブジェクトファイルを“カレント実行ディレクトリ”に作成します。

具体的にカレント(データ)ディレクトリにあるprog.cというCのプログラムをコンパイルしたい場合は，

```
cc prog.c
```

と入力すればOKです。なおコンパイルドライバを呼ぶときに，Cのプログラムはもちろん，アセンブリ言語のプログラムやリロケータブルモジュール形式のものも，その名前を指定することで自動的にアセンブルやリンクが行われ，ちゃんとひとつのオブジェクトが生成されます。

このへんはコンパイルドライバ様々といったところです。なおこの場合，プログラムのサフィックス(拡張子)が“.c”だとCのソースファイル，“.a”だとアセンブリ言語のソースファイル，“.r”だとリロケータブルモジュールのファイルであると自動的に判断されます。

さて，先ほど“カレント実行ディレクトリ”という言葉が出ましたが，これはOS-9特有のもので，前回，前々回で説明したはずですので，その話を思い出してください。

コンパイルした結果はどうかというと，このCコンパイラは結構，効果的なコードを生成するようで，なにもしなくてもポジションインディペンデント，リエントラント，ROM化可能なコードなどが生成できます。このあたりはさすがOS-9といったところでしょうか。

さて，コンパイル速度についてですが，極端に速いということはないのですが，XCに比べるとかなり速いといえます。

最後にXCやUNIXのCからの移植を考えてみましょう。このときにもっとも障害になると思われるのは，コンパイラのオプションがそれらのどれとも異なっていることと，コンパイル時にファイルの判断の基準となるサフィックス（Human68kでは拡張子）が異なるということです。

場合によっては識別する変数名の長さが移植の際の問題になる場合もあります。OS-9用のCでは変数名は256文字まで認識されるので，KCL(Kyoto Common Lisp)のような異常に長い変数名を使ったプログラムでもそれほど手を加えずにコンパイルできるかもしれません。

多くのソースプログラムをコンパイルするときなどには必需品となるユーティリティにmakeがあります。Human68kのXCの場合は標準的なmakeがありませんので，あまり影響はないかもしれませんが（ネットなどでは流れているが，あまりいいという噂を聞かない），UNIXからの移植の場合，makeを使ってコンパイルなどをするものはmakeファイルを書き直す必要があるものと思われます。逆にいえばこれを書き直せば，移植作業は非常に楽になるのです。OS-9/X68000ではmakeはCの標準装備です。

その他

OS-9/X68000のCには別売りですがソースレベルデバッガもあり，これもあわせて使えばかなりプログラミングもやりやすいものと思われます。また，Human68k側からのファイルの転送はOS-9/X68000付属のhufileというコマンドで簡単にできます。Human68kのファイルが読めるということは，MS-DOSのファイルが読めるということで，ファイルのコンバートの心配は不要でしょう。

なんだかんだといいながら，このCは58,000円と，比較的安価なものとなっています。XCが驚異的低価格で発売されていますが，オマケでmakeコマンドとスクリーンエディタが2つ（SCREDとKuMACS）ついてくるのですから，コストパフォーマンスではそれほど負けていません。最近のCではソースデバッガがついてくることが多いようなので，あとは別売りのソースレベルデバッガの値段に期待してみたいところです。

リスト2　こんな変数名も大丈夫

```
==================  valtest.c  ==================
 1: /*
 2:     Val Test Sample
 3: */
 4:
 5: #include <stdio.h>
 6: main()
 7: {
 8:  int a1234567890123456789012345678901234567890123456789012345678901234567890a,
 9:      a1234567890123456789012345678901234567890123456789012345678901234567890b;
10:
11: a1234567890123456789012345678901234567890123456789012345678901234567890a = 1;
12: a1234567890123456789012345678901234567890123456789012345678901234567890b = 0;
13:
14: printf("%d¥n",a1234567890123456789012345678901234567890123456789012345678901234567890a
);
15: }
```

ニホン語，不得意

満開製作所 祝 一平
Iwai Ippei

今月のタイトルを見て，「ハテ? ついに日本語処理でもやるのかな」と思ってしまった方，ハズレではありませんが，そこまでの道のりはちょっと遠いよーです。まずは，Cのなかに潜んでいる論理式の複雑さをねちねちと解説してくれるようなので，基本構造を真面目に学んでみてください。

まずは，先月残してあった宿題である。アセンブラのソースプログラム中で，「:」を使うことによって，マルチステートメントを可能にするフィルタを作れということであった。これは要するに，「:」があったら，そこで2行に分割するということなわけだ(ただし文字列中や，ラベルの右端は除外する)。

で，実はわざと書かなかったのであるが，このように「検索されるパターンのなかに“¥n”(改行コード)を含まない」ことが明白な場合は，

行単位で処理する

と，とってもお徳用なのである。ただし，そのためには1行の長さがだいたい決まっていて，とんでもなく長い行(たとえば数100バイト)などはない，という条件も必要となってくる。ここでは，扱うのはアセンブラのソースプログラムであるから，1行の長さもたかだか100バイト程度であろうと判断される。てなわけで，このような場合は行単位で処理すると楽なのである。解答例がリスト1の「kirihana.c」である。

プログラムの大まかな流れは，

1) 標準入力から1行(改行コードがくるまで)読み込む
2) その1行について処理する

というのを，ファイルエンドになるまで繰り返すわけである。

ところで，作っているうちにわかったのだが，AS.Xでは，ラベルとして，英数字以外にも，「_」や「￣」(アンダーバーやアッパーバー)が使えるのである。もしかしたらほかにも使えるのかもしれない。また，先月号の桒野氏の記事にもあったように，ラベルをスペースやタブで字下げすることも可能なのである。それらに対応するために，プログラムはちょっと複雑になってしまった。具体的には，最初に行の先頭にあるスペースとタブは無条件で通してしまう(18，19行)。次に続く英数字も無条件に通してしまう(28~39行)。そうして英数字以外のキャラクタに出くわしたら，いよいよ本格的にチェックの始まりである(41行)。肝心なのは，そのすぐ先にあるswitch文なわけだ。「'」もしくは「"」で，文字列データが始まったらその文字列が終わるまで素通ししてしまうのが43~51行。そのあとにあるのが，「:」にぶつかったときの処理で，「:」の代わりに，改行コードとタブをputcharしている。

問題はその次の「*」の処理である。ここでは，

s[−1]=直前の文字がスペースもしくはタブだったら，この「*」は，コメント開始の印である

と解釈し，それ以降の1行をflush(つまり，全部素通し)している。ここはなかなかに問題なところなのである。つまり，

```
move.1  #10*3, d0
```

なんてプログラムがあったら，「*」の前後にはスペースを入れてはいけないのである。ま，その場合でもおかしなことになるのは，極めて限られた場合だから，それほど大きな問題ではないだろうと，タカをくくってしまうのである。

あとちょっと毛色が変わっているのが，「peekc」であろう。これに渡す引数は，「ポインタへのポインタ」である(どーだ，頭が混乱してきただろう)。この関数の機能は，「そのポインタが指している文字を返す。もしも全角文字の場合は2バイト返す。そして その際，ポインタを自動的にそのバイト数分だけインクリメントする」である。

たとえば，

```
s→“あいう”
```

だったなら，

```
c=peekc(&s);
```

を実行したあとでは，cは82A0H(つまり「あ」のシフトJISコード)になっており，sは(勝手に)+2されているのである。よって，その次にまた

```
c=peekc(&s);
```

としたならば，今度は，cは82A2H(つまり「い」のシフトJISコード)になっており，sはさらに+2されているのである。よーするに，これは「2つの値を返す関数」なわけだな。「&演算子」という奴は，このように使ったりもするのである。で，これは比較的上級なテクニックなので，少し考えてわからなかったら，さっさと諦めて機が熟するのを待つよーに。

さて，プログラムの使い方は，

```
kirihana < 元ファイル > 先ファイル
```

である。

バッチファイルで，XAS.BATなんてのを次のように作って用意しておくと便利であろう。

```
kirihana < %1 > temp000.s
as temp000 /o %1.o
lk %1
```

* * *

リスト1にある，1行を読み込む関数getlineであるが，これは『K&R』の31，73ページに載っているものである。この関数に渡す引数は，sとlimである。

sはchar型へのポインタで，「読み込んだ文字列を，ここへ格納

して返してね」ということになっている。

limはint型で，最大文字数である。なんでこんなものが必要かというと，sの指す領域に何文字まで格納できるかが問題だからなのだ。つまり，「もしも1行がこれ以上の長さになる場合は，とりあえず途中で打ち切ってください」なのである。さもなくば，なにかの間違いで異常に長い行を読み込んだときに，ほかのデータエリアやスタックを破壊してしまって，最悪の場合は暴走ということになってしまうのである。そういうことを防ぐために，いちいちリミットを指定しておくのである。もちろん途中で打ち切った場合，残りの部分は次に呼ばれたときに返してくれることになっている。なお，関数自体が返す値は返す文字列の長さである。

さて，このgetlineであるが，こいつはwhile文のなかの条件式がちょいと食わせものである。というのは，Cにおける論理式(条件式）という奴は，なかなかに特殊なのである。で，今月は，このgetlineの解剖を，ねちねちとやってみることにする。こんな小さな関数でも，結構奥は深いのである。

退化してみる

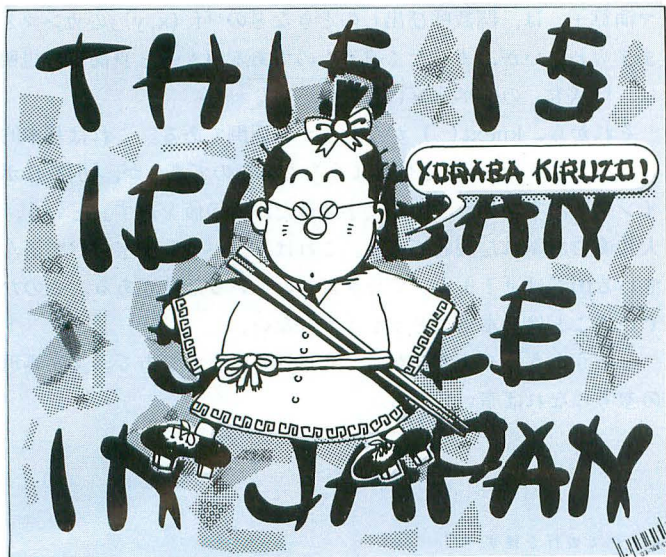
まず，getlineをうんとダサく書き直してみる。これは，いわば，もっとも「Cらしくない」Cのプログラムである。それがリスト2の「kakinao.c」である。関数としての機能は，もちろん同じままである。これはある意味では，BASIC的に書き直しているのでもあるわけだ。

Cという言語の特徴のひとつに，「論理式は値が決定したらその先は評価しない」というのがある。これはどういうことかという

リスト1 kirihana. c

```
  1: #define MAXLINE 1000
  2: #define EOF     -1
  3:
  4: main()
  5: {
  6:         char line[MAXLINE];
  7:
  8:         while(getline(line,MAXLINE) > 0) {
  9:                 doline(line);
 10:         }
 11: }
 12:
 13: doline(s)
 14: char *s;
 15: {
 16:         int c,d;
 17:
 18:         while(((c = peekc(&s))==' ') || ( c=='¥t'))
 19:                 kputchar(c);
 20:
 21: /* c = first data */
 22:         if (c=='*') {   /* 1 line comment */
 23:                 kputchar(c);
 24:                 flush(s);
 25:                 return;
 26:         }
 27:
 28:         while(c != '¥0') {
 29:                 kputchar(c);
 30:                 if(!((('0'<=c)&&(c<='9')) ||
 31:                      (('A'<=c)&&(c<='Z')) ||
 32:                      (('a'<=c)&&(c<='z')) ||
 33:                      (c=='~') ||
 34:                      (c=='_'))) {
 35:                         c = peekc(&s);
 36:                         break;
 37:                 }
 38:                 c = peekc(&s);
 39:         }
 40:
 41:         while(c != '¥0') {
 42:                 switch(c) {
 43:                 case '¥'':
 44:                 case '¥"':         /* 文字列データの開始 */
 45:                         kputchar(c);
 46:                         while((d = peekc(&s)) != '¥0') {
 47:                                 kputchar(d);
 48:                                 if (d==c)
 49:                                         break;
 50:                         }
 51:                         break;
 52:                 case ':':          /* コロン！ */
 53:                         putchar('¥n');
 54:                         putchar('¥t');
 55:                         break;
 56:                 case '*':
 57:                         putchar(c);
 58:                         if ((s[-1]=='¥t') || (s[-1]==' '))
 59:                                 flush(s);
 60:                         break;
 61:                 default:
 62:                         kputchar(c);
 63:                         break;
 64:                 }
 65:                 c = peekc(&s);
 66:         }
 67: }
 68:
 69: kputchar(c)
 70: int c;
 71: {       if (c & 0xff00)
 72:                 putchar(c>>8);
 73:         putchar(c & 0xff);
 74: }
 75:
 76: peekc(p)
 77: char **p;
 78: {
 79:         int c,c1;
 80:         c = *((*p)++);
 81:         if (iskanji(c)) {
 82:                 c1 = *((*p)++);
 83:                 if (iskanji2(c1))
 84:                         return((c<<8) | c1);
 85:                 else {
 86:                         (*p)--;
 87:                         return(c);
 88:                 }
 89:         } else return(c);
 90: }
 91:
 92: getline(s, lim)
 93: char s[];
 94: int lim;
 95: {
 96:         int  c, i;
 97:
 98:         i = 0;
 99:         while (--lim > 0 && (c = getchar()) != EOF && c != '¥n')
100:                 s[i++] = c;
101:         if (c == '¥n')
102:                 s[i++] = c;
103:         s[i] = '¥0';
104:         return(i);
105: }
106:
107: flush(s)
108: char *s;
109: {
110:         while(*s)
111:                 putchar(*s++);
112: }
```

リスト2 kakinao. c

```
 1: getline(s, lim)
 2: char s[];
 3: int lim;
 4: {
 5:         int  c, i;
 6:
 7:         i = 0;
 8:
 9: loop:
10:         lim = lim-1;
11:         if (lim > 0) {
12:                 c = getchar();
13:                 if ((c != EOF) && (c != '¥n')) {
14:                         s[i] = c;
15:                         i = i+1;
16:                         goto loop;
17:                 }
18:         }
19:
20:         if (c == '¥n') {
21:                 s[i] = c;
22:                 i = i+1;
23:         }
24:         s[i] = '¥0';
25:         return(i);
26: }
27:
```

と，たとえば，

```
if (△△ && □□) {
      ⋮
} else {
      ⋮
}
```

なんてプログラムがあったとする。普通のコンピュータ言語ではこういう場合は，

1) △△を計算する
2) □□を計算する
3) ふたつの値を論理的ANDして，もしも真だったら〜
      ⋮

という手順で進んで行くことになるであろう。たとえば，BASICでちょっと複雑なプログラムを作ったことがある人なら，こんなのに出くわしてムカついた経験があるかもしれない(私はある！)。

```
DIM A (100)
  ⋮
IF I<=100 AND A(I)=1 THEN ~
```

先に，配列宣言で「DIM A(100)」としているので，それをチェックするためにIF文の条件のなかで「I<=100」としているわけである。こうしておけば，Iは100を超えることはないので，エラーは出ないというふうに思えるであろうが，実はこれではだめなのだ。つまり，もしもI=101だったとすると，

I<=100 → 偽

と出たあとで，なんとマヌケなことに「A(I)=1」，すなわち「A(101)=1」の真偽を判定しに行くのである。AND の片一方が偽なんだから，もう片方がなんであっても必ず偽になるはずである。ああ，それなのにそれなのに，わざわざ「A(101)」を計算し，そしてエラーで止まるのである。だから，BASICでは，

```
IF I<=100 THEN IF A(I)=1 THEN ~
```

と，2段重ねのIF文にしなければならないわけである。もしもこれがちょいと複雑だったりすると，たちまち泥沼になってしまうのである。

ところが，Cではここが**まったく違う**のである。すなわち，

1) △△を計算する
2) もし△△が偽（=0）だったら，elseの処理に向かう
3) □□を計算する
4) もしも□□が真（<>0）だったら，次（then）の処理に向かう。偽だったらelseの処理に向かう

となっているのである。

よって，Cでは，上の例であれば，

```
if ((i<=100) && (a[i]==1)) ~
```

とすれば，iが101だったら，a [i]は評価しないのである。これは，なかなかの気配りなのであるが，しかし，反面，複雑なプログラムだと，if 文を追いかけるのがちょっとしんどくなってくるという欠点もある。

なお，これはOR（Cでの記号だと｜｜）でも似たようなことがあって，「片方が真だったなら，もう片方は計算されない」のである。これは，ORの片方が真だったならば，もう片方がなんであっても結果は真だからである。Cではこのようなことによって処理の高速化が図られているのである。

さて，ここで問題である。もしも，

```
((△△ && □□) || ○○)
```

なんてのがあって，△△が偽（=0）であったら，□□や○○は，はたして計算されるのだろうか？

これは，各自で工夫し，実際にチェックのためのプログラムを組んでみて，確かめてもらいたい。

で，リスト2では，ここらへんのことが如実にわかるように書いたつもりである。

このように，Cでは何気なく見えるところに，意外な意味が潜んでいたりするのである。BASICとは，水と油ほども違う発想で作られているのだ。

呪われた言語

kirihana. cでは，問題にはならないが，多くの文字列処理をするプログラムでは，本当は正しく日本語を使えなければいけないのである。そしてそのためには，そこらあたりをちゃんと考慮してプログラムをしなければならないのである。さもないと，ずっこけることになる。

はたして，具体的にどんなときにまずいかというと，たとえばHumanやMS-DOSで漢字を扱うときに使われているシフトJISコードでは，全角のスペースは81_H，40_Hの2バイトで表されているのである。で，困ったちゃんなのが，2バイト目の40_Hなのだな。というのは，こいつは ASCII コードでは半角の「@」だからなのだ。だもんで，単純に半角の「@」をサーチしたりすると，あにはからんや，全角のスペースが引っかかってくるわけである。そこで，お箸の国の人たちは，文字を扱うときには2バイト単位で勝負しなければならないわけである。

そういうわけで，いきなり出てきたのが，リスト3の「sagasida. c」である。これは『K&R』の72〜73ページにある，文字列検索プログラムの日本語版である。kindex( )，kgetchar( )，kputchar( )，knext( )などの関数が，改造/追加されている。『K&R』

の日本語に対応していないパターン照合関数のindex( )とも比較してほしい。ひとつ注意しておきたいものが，kindexのなかの2つ目のfor文のなかである。よく見ると，

j=i，k=0；

となっている。これにはカンマ演算子などという，偉そーな名前が付いているが，簡単にいってしまえば，「結果の値をチャイする」という演算子である。

だもんだから，たとえば，

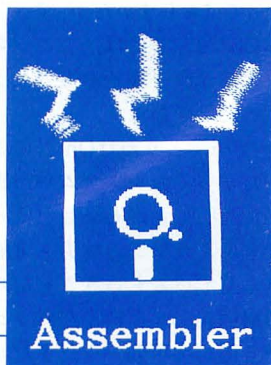
i=1，2，3；

なんてこともできて，この場合は「1捨てて，2も捨てて，んでもって3をiに代入する」になるわけである。なお，この「カンマ演算子」は，関数呼び出しのときなどの「f (x, y)」のカンマとまぎらわしいが，まったく違うものである（もっと真面目な説明は，『K&R』の64ページに載っている）。

それから，knext( )がなかなかに面倒である。これは基本的にはリスト1のpeekcと似たようなものなのであるが，「引数にポインタと添字を与えて，1文字進めた添字の値を返す」という，大和魂にあふれた関数である。これは，1バイト文字だけで済んでいる国では「i++」でカタがついている部分である。このたぐいのことは，本当にどうしようもない。

というようにグチりながら，今月はこれまでである。文字処理の参考になれば幸いである。

リスト3　sagasida. c

```
 1: #define MAXLINE 1000
 2: #define EOF     -1
 3:
 4: main(argc,argv) /* あるパターンと一致するすべての行を捜す */
 5: int argc;
 6: char *argv[];
 7: {
 8:         char line[MAXLINE];
 9:
10:         while(getline(line,MAXLINE) > 0)
11:                 if (kindex(line,argv[1]) >= 0)
12:                         printf("%s",line);
13: }
14:
15: getline(s, lim) /* 行をsに入れ、長さを返す */
16: char s[];
17: int lim;
18: {
19:         int  c, i;
20:
21:         i = 0;
22:         while (--lim > 0 && (c = getchar()) != EOF && c != '¥n')
23:                 s[i++] = c;
24:         if (c == '¥n')
25:                 s[i++] = c;
26:         s[i] = '¥0';
27:         return(i);
28: }
29:
30: kindex(s,t)     /* tの添字をsに返す、なければ－1を返す */
31: char s[], t[];
32: {
33:         int i,j,k;
34:
35:         for (i = 0; s[i] != '¥0'; i = knext(s,i)) {
36:                 for(j=i,k=0;t[k] != '¥0' && (getk(s,j) == getk(t,k));j=k
next(s,j),k=knext(s,k))
37:                         ;
38:                 if (t[k] == '¥0')
39:                         return(i);
40:         }
41:         return(-1);
42: }
43:
44: getk(s,i)       /* 全角か半角かを判定し、1文字を返す */
45: char *s;
46: int i;
47: {
48:         int c,c1;
49:
50:         c = s[i++];
51:         if (iskanji(c)) {
52:                 c1 = s[i];
53:                 if (iskanji2(c1)) {
54:                         i++;
55:                         return((c<<8) | c1);
56:                 } else
57:                         return(c);
58:         } else return(c);
59: }
60:
61: knext(s,i)      /* 全角か半角かを判定し、iを＋1もしくは＋2する */
62: char *s;
63: int i;
64: {
65:         if (iskanji(s[i++]) && iskanji2(s[i]))
66:                 return(++i);
67:         else
68:                 return(i);
69: }
70:
71: kputchar(c)     /* 全角か半角かを判定し、ｐｕｔｃｈａｒする */
72: int c;
73: {
74:         if (c & 0xff00)
75:                 putchar(c>>8);
76:
77:         putchar(c & 0xff);
78: }
79:
```

X68000 新連載予告編 マシン語プログラミング入門

Oh!Xでは，いよいよ来月からX68000のアセンブラによるプログラミング講座を開始することになった。筆者は「Z80マシン語ゲーム工房」で高密度プログラミング講座を完結させた村田敏幸氏が，X68000にフィールドを移して長期連載に挑戦する。

Murata Toshiyuki
村田 敏幸

金色のフロッピーディスク

男は過って大事なディスクをクラッシュしてしまった。どうやっても修復できそうにないことを知ると，彼は大きくため息をつき，未整理のまま積まれたディスクの山を振り返った。

このどこかにバックアップディスクがあるはずなのだ。が，ラベルも貼られていない数十枚のディスクのなかから，必要な1枚を探し出すことは彼にとっては不可能に近いことだった。

と，突然，0番ドライブが音もなく，いや，馴染みの音を立てて，1枚のディスクをイジェクトした。それは，燦然と輝く金色のディスクであった。と同時にディスプレイには65536色でデジタイズされた白髪の老人が映し出された。

「わしは神ぢゃ」

老人は簡潔に自己紹介すると，言葉をついだ。

「おまえの探しているのは，インタプリタならではの対話性のよい環境に加えて，Cライクな制御構造を持ち，複数行にわたる関数の記述もでき，さらには外部関数によりグラフィックもサウンドもお手のものというX-BASICが入った，この金のディスクかね？」

「いいえ，神様。僕が探しているのは，そのような便利で立派なディスクではありません」

「そうか」

と，神は金色のディスクを引っ込めると，今度は鈍い銀の光を放つ2枚組のディスクと分厚い4冊のマニュアルを示して，こう言った。

「では，おまえが探しているのは，低レベルな処理から高レベルな処理まで自在にこなし，巷ではできる人からできない人まで人気街道驀進中のC言語の雄にして，ANSI Cに準拠し，膨大な数のライブラリによりX68000の機能を隅から隅まで活用でき，総計2000ページに及ぶマニュアルがついて，サンキュッパの持ってけ泥棒価格，発売以来大好評，1年余りで1万本が売れたというC compiler PRO-68Kが入った，この銀のディスクかね？」

「いいえ，神様。僕が探しているのは，そのような贅沢で美しいディスクではありません」

神は少し驚いた顔をしたが，すぐに平静さを取り戻し，今度はくたびれたノーブランドのディスクを取り出した。

「ならば，おまえの探しているのは，このディスクかね？」

「それです」

「正直なやつぢゃ。しかし，このディスクに入っているのは，でき上がったプログラムはコンパクトで速いが，CPUの命令と1対1で対応するという低レベルさゆえに，些細な処理にも数ステップを要し，開発に時間がかかるし，小さいミスで暴走はするし，デバッグにも手間暇と野性的な勘が必要とされるというマシン語プログラムを作成するためのツール，アセンブラとリンカぐらいのものぢゃが」

「でも，好きなんです。マシン語じゃなきゃ，ヤなんです」

「うーむ，もしかして，おまえは68000のマシン語は直交性に優れ，人によっては高級言語のようだという言葉を鵜呑みにしているのではないか？」

「さすがに高級言語というのは言い過ぎだと思っています。直交性うんぬんというのも，Z80などの8ビットCPUやほかの16ビットCPU……8086のことですが……よりはマシ程度に捉えています。ところで，用が済んだら早いとこ，ディスプレイから消えてくれませんか。僕は来月から始めるX68000のマシン語プログラミングに関する連載の準備をしていたところなんです。あなたがずっとそこにいると，作業がぜんぜん進まないんですよ」

「神に向かってなんたる口のききよう。しかし，正直者には褒美を与えなければなるまいな。4Mの増設RAMボードなり，ハードディスクなり，おまえの好きなものを言うがよい」

村田敏幸は少し考えてから答えた。

「では，LK.Xの2倍ぐらい速いリンカをいただけますか？」

【物語の解説】

これは昔から有名な，ある正直なマシン語プログラマのお話である。しかし，彼が金のディスクを拒否したのは，けっして欲がなかったからではないだろう。彼の本音を言えば，探していたのは，

「変数宣言が必要といいながらINTの変数だけは無宣言で使えるという余計なお世話のためにあらぬバグを生んだり，関数と配列の受け渡しができなかったり，LOAD/SAVEが遅くてイライラしたり，エラー時やBREAK時にローカルな変数を保存していてくれないので関数単位のデバッグが困難だったり，マニュアルにも不備が目立ってしょうがないX-BASIC」

ではなく，また，

「ANSI準拠といいたって，まだ規格案が固まらない時代のシロモノで，ろくなオプティマイズもしないで大きなコードを平気で吐くわ，当然速度も遅いわのXC」

でもなかったのだろう。

確かにX-BASICもXCもまだまだ完璧ではない。なんといってもX-BASICはCへのコンバートを前提に作られた世界初の試みだったのだ。XCにしても，日本のパソコンでは初めてハードウェア固有の機能をサポートしたCであり，しかも生まれてから1年ちょっとしかたっていない。生まれたてのコンパイラなんてのは，こんなものだ。MS-CやTurboCだって世界の86ユーザーの頂点に立つ人々が長年の努力の末に完成したものだ。XCにしてもこなれるには，もうしばらく時間がかかるだろう。

だが，彼にとっての問題はほかにある。そもそも彼は，どんなに便利であっても，どんなにお手軽であっても，他人が用意してくれたソフトの上であぐらをかくことのできない純粋なプログラマだからだ。

*　　　　　*

さて，X68000は実力あるマシン語プログラマを必要としている。少なくとも今後はもっと必要とされるだろう。もちろん，プログラマの供給が追いつかず，X68000自体がすたれてしまってはおしまいだが，そうならないためにもOh!Xとしては，ここいらで本格的なマシン語講座を開始しなければという判断に至ったのである。

X68000のユーザー，とりわけOh!Xの読者にはうるさがたが多く，システムや市販のソフトウエアに対しても厳しい目を持っている。X68000というマシンの能力と可能性をそれだけ高く見ているということだろう。だが，このMPU68000を積んだパソコンが登場してから僅か2年弱，まだまだユーザーの力はX68000の能力をフルに引き出すところまで至ってはいない。

ユーザーはメーカーやソフトハウスに対し，よりよいシステム，よりよいアプリケーションを要求する。だがそれらを開発するプログラマもまたユーザーのなかから生まれてくるものなのだ。

そして，X68000で培われた技術は，そのまま68020/68030へと受け継がれる。数年後X68030のC compiler PRO-68K Ver.Xを作るのは君かもしれないのだ。

というわけで，今月は読者のみんなと僕の準備期間ということで，予告編をお届けしよう。

マシン語のすすめ

マシン語プログラムは実行速度が速い，なぜなら，マシン語はコンピュータが理解し，直接実行できる唯一の言葉であるからだ，なぁんて話はOh!Xを続けて読んでいる人なら誰でも知っていることだ。この速さに惹かれてマシン語を覚えようと思った人は多いし，実際，高速性はマシン語の利点の第1番目に挙げられるだろう。

また，プログラムを小さく作ることにかけても，マシン語の右に出るものはいない。X68000ならメモリが最低でも1Mあるからそれほどプログラムの大きさを気にする必要はないように思えるところだが，フロッピーディスクの容量を考えれば，そうのんびり構えているわけにもいかないだろう。あー，64Kバイトの大容量メモリに320Kバイトのフロッピーディスクが1基あれば天国だった時代が懐かしい。

さらに，小回りのよさという点も見過ごせない。痒いところに手が届く，といってもいいし，なんでもあり，といってもいい。高級言語では，ハードウェアに密着した処理を行うことが困難だったり，不可能な場合さえあるが，マシン語を使えば，そういった制約なしにコンピュータの性能を隅隅まで生かすことができるのだ。中森さんのX68000BASIC入門の連載でも，マシン語による外部関数を作っていたのを思い出してもらいたい。

さて，「Cなどのコンパイラ言語を使えば，労せずしてマシン語のプログラムを作ることができるんじゃないの？」という声が聞こえてきそうだ。特にCはコンパイラ言語にしてはプログラムをコンパクトにまとめることができ，比較的小回りもきく処理系として知られている。でも，最初からアセンブラで書いたプログラムに比べれば，まだまだ，遅いのだ，でかいのだ，まどろっこしいのだ。

プログラマの選択

もちろん，アセンブリ言語にも欠点はある。1つひとつの命令の機能が低いので，ごく単純な処理を実現するのにも，いくつもの命令を組み合わせて記述しなければならないのだ。単独の命令でできることといえば，データの転送（代入）とか，整数の加減算（できたとしても乗除算）程度の簡単な計算，あとは分岐（BASICでいうGOTO）やサブルーチンコール（GOSUB）などの制御ぐらいのものだ。BASICやCなどの高級言語では1行で書けるような処理も，マシン語で書けば，あっという間に数～数10命令に膨れ上がる。このことは，そのまま開発時間・手間に跳ね返ってくるわけだ。

ここで，プログラマは大きな選択を迫られることになる。すなわち，速度とオブジェクト効率の究極を目指して，その代わり，余計な手間暇をかけてアセンブラで開発するか，そこそこの手間，そこそこのコンパクトさ，そこそこの速度で満足し，高級言語を使うか，だ。

結局のところ，どちらを選択するかは人それぞれの問題だし，ケースによってもどちらが有利かという問題は変わってくる。

とりあえずBASICやCで作っておき，必要な部分だけアセンブラで作り直すといったことは多くの一般のユーザーにとって有効な方法だろう。またアセンブラの知識があれば，Cで記述する場合でも，コンパイラの吐き出すコードをチェックして少しでもよいコードを作る記述方法を模索することも可能となる。

いずれにしてもアセンブラによるプログラミングを修得しておくことは非常に大きな力となるということだ。

カリキュラムのお知らせ まずはDOSコールを利用して

来月から始まる連載では，Human68k上でプログラミングを扱い，OS-9/X68000，CP/M-68Kは無視する。Human68kはX68000を持っている人ならだれでも持っている

OSだというのが，その主な理由だ。

で，今後の予定だが，まずはHuman68kに用意されているサービスであるDOSコールを使って，文字の入出力だとか，ファイルの取り扱いといった話から始めてみたい。最初の題材は簡単なフィルタかな。あとは小規模な外部コマンドをいくつか作ってみることになるだろう。

ぐらふぃっくピカピカ，すぷらいとギュンギュン，さうんどピポパポ，FDCガシガシ，といった派手な話を期待した読者諸君には申しわけないけど，どんなプログラムでも大なり小なり入出力の処理は必要不可欠なものだから，ここをがっちり押さえておくことは意味のあることだと思う。楽しみはとっておこうじゃないか。

DOSコールを使う利点は入出力の処理を手軽に，そして，統一的に記述することができることにある。感じとしてはX-BASICやXCにあるファイル処理関数と同じようなものだと思っていい。DOSコールを使えば，マシン語の命令をそれほどたくさん覚えなくても，それなりのプログラムが作れてしまうから，入門者はこの時期にマシン語プログラミングの雰囲気を掴んでもらいたいものだ。

その次のステップでは，ファイル処理という制限を外して，X68000の豊富な機能で遊んでみるわけだ。グラフィックでしょ，スプライトでしょ，AD PCMにFM音源でしょ，そのほかいろいろ！　これらの機能はみんなIOCSコールによってサポートされているから，ハードウェアの細かいことを知らなくても使えてしまうというのが，ありがたい。場合によってはIOCSを使わず，直接ハードウェアにタッチするのもありだな。

まぁ，どこまでできるかわからないけど，あわてず，さわがず，のんびりと，そして確実にX68000を掘り下げていけたらいーな，と思っている。猿がらっきょうの皮をむくように，あとにはなにも残らないというのが理想だね（女の子の服を優しく1枚1枚……というフレーズも考えたんだけど，「一番おいしいところを残すの？」と祝さんに突っ込まれそうな気がしてやめた）。

マシン語開発ツール これだけは準備してね

ここまで読んで，ふつふつと血がたぎるのを覚えた人，なーんとなく面白そうだと思った人は，さっそくつぎの準備にとりかかってもらいたい。マシン語プログラミングにはなにが必要なのか，そんな話をしておこう。

マシン語プログラミングといっても，それほど特殊なものは要らない。X68000が1台以上と，あとはいくつかのツールがあればいい。ツールとしては，ソースプログラムを作成するためのテキストエディタ，ソースをマシン語に変換するアセンブラ，そして，プログラムを分割して開発するときに必要なリンカ，この3つだな。

●エディタ

X68000で使えるテキストエディタには，本体付属のED.X，市販されているWINDEX（定価28,000円，ジェー・イー・エル），Final（定価38,000円，エー・エス・ピー），PDSで出回っているmicroEMACS（基本的に無料，ネットなどで入手可能）なんかがある。テキストエディタはプログラムを作成するときに一番長い時間使うツールであり，エディタがどれだけ手に馴染むかが，プログラム開発をスムーズに進める鍵になる。こだわりを持って選びたいものだ。

どれがいいかなんて聞かないでほしい。エディタほど人の好みが分かれるものはないんだから。だれでも自分の使い込んだエディタが一番使いやすいといい，そのくせ，不満もそれなりに持っているものなんだ。ちなみに僕はED. Xを使っている（あれは，ごくふつーのスクリーンエディタだ）けれど，時間があったら自分専用のを作りたいと思っている。1カ月か2カ月，完全にフリーの時間がとれれば，すぐにでも作るんだけどな。

●アセンブラ，リンカ

X68000のアセンブラとリンカとして標準的なのはAS. XとLK. Xだ。AS.Xは分割アセンブルにも対応した大規模プログラミングにも向く高級なアセンブラであり，ソーステキストをアセンブルし，オブジェクトファイルを生成する。分割アセンブルをするときには，このオブジェクトファイルをつなぎ合わせてひとつの実行形式ファイルを作成するリンクという作業を行う必要があり，それを担当するのがリンカであるLK. Xというわけだ。もっとも，AS. Xの場合は(AS.Xに限ったことではないが）プログラム作成の手順を統一するために，分割アセンブルをしない場合でもリンカを通すことになっている。この場合，リンカは単に実行形式ファイルを作るという働きをすることになる。その意味で，リンクまでまとめて「アセンブルする」ということがあるので混乱しないように。

元祖X68000（CZ-600C）のシステムには福袋というディレクトリがあって，その中にこっそりとAS. XとLK. Xが入っていたのだが，ACE以降では残念ながら削られてしまったようなので，手に入れるには市販されているものを買ってくるしかない。「THE福袋V2.0」(9,980円，シャープ）か，「C compiler PRO-68K」(39,800円，シャープ）のどちらかが必要だ。

THE福袋V2.0にはAS.XとLK.Xに加えて，マシン語プログラムのデバッグを助けるデバッガDB. X，X-BASICのVer. 2.00や便利なコマンドがいくつか，それから，アセンブラなどの使い方やモトローラ MPU68000の命令の解説などが載っている「アセンブラマニュアル」と，Human68kのDOSコール，X68000のIOCSコールの一覧や，X-BASICの外部関数の作り方，デバイスドライバの作り方などなどの情報がまとめられている「プログラマーズマニュアル」が入っている。これで9,980円というのは安い（MS-DOS用のアセンブラ MASM は確か40,000円だったと思う）。マシン語を志す人には無条件でお勧めできるアイテムだ。

僕がアセンブラにこだわるわけ

熱力学の話ではないが，「ほんとは誰でも楽をしたい」の法則により，多くの人はCなどのコンパイラ言語へと流れていく。

ところが，時代に逆らって「ヤだ，ヤだ，ヤだ。マシン語じゃなきゃヤだ」と意地を張る，僕のような人間もいるわけだ。僕自身，それがなぜなのかは，あまり考えてみたことがない。はっきりしているのは，単に速度やプログラムの大きさの問題ではないということだ。強いていうなら，自分のマシンを直接ドライブすることに魅力を感じているのかもしれない。

マシン語プログラムは，CPUの命令と1対1に対応する簡単な略語からなるアセンブリ言語を用いて開発される。このアセンブリソースプログラムをアセンブラというプログラムに通すことで，実行可能なマシン語プログラムができ上がる。手順そのものはコンパイラ言語での開発とほとんど変わらないし，最終的にマシン語プログラムになるのも表向き同じだ。が，コンパイラがソースプログラムをいくつもの命令の組み合わせに「翻訳」するのに対して，アセンブラはアセンブリソース上の1命令をマシン語の1命令に単純に「変換」するだけだから，ソースプログラムの1命令に対するプログラマの思い入れ（のようなもの）は，かなり違ってくる。

自作のマシン語プログラムが走るとき，CPUの動作は完全に僕の手の内にある。どの瞬間を切り出してみても，CPUは僕がプログラムに書いたとおりの動作をし，そして，それ以上のことはなにもしない。言葉ではどうもうまく言えないけれど，僕はアセンブラプログラミングのそんなところに惚れちゃってるのだ。

また，C compiler PRO-68Kには，福袋V2.0がそっくりそのまま含まれている。もちろんメインはCコンパイラとそのライブラリ一式（部分的にはマシン語プログラムから直接利用することもできる），そしてライブラリのソースリスト（アセンブリ言語で書かれているから，マシン語学習にももってこい）に，「Cリファレンスマニュアル」と「ライブラリマニュアル」が含まれる。

というわけで，財布の中身と相談して，どちらかを手に入れてもらいたい。また，元祖のユーザーにとっても，プログラマーズマニュアルはおいしいし，デバッガはあると便利だし，アセンブラやリンカもバグとりと思われる若干のバージョンアップが行われているので，福袋V2.0ぐらいは買っておいても損はないと思う。

なお，最近サードパーティから「CMA68K」（29,800円，シティソフト）というアセンブラが発売された。これはAS.Xの上位コンパチの機能を備えたアセンブラだが，使用するにあたっては，後日同社から発売されるリンカか，LK.Xが必要だ。

●環境設定

こうして必要なプログラムが揃ったら，プログラミング開発環境を整えておこう。まず，アセンブラやエディタなどはひとつのディレクトリにまとめておきたい。BINに突っ込んでおけばいいだろう。また，ハードディスクを持っている人は作業用のディレクトリを作っておくといい。ハードディスクがなければ，作業用のシステムディスクを1枚作って，必要なツールや作りかけのプログラムを入れておくのを勧める。

今度の連載では，特別なデバイスドライバの組み込みなどは必要としない予定でいるから，その辺はあまり気にすることはない。ただ，日頃ビジュアルシェルしか使っていない人には悪いのだけれど，作業はCOMMAND.X上で行うことになるので，VS.Xははずしておいてもらいたい。このとき，VS.XからCOMMAND.Xを起動してもよいのだが，場合によってはメモリ不足が起きることが考えられるから（メモリを増設すれば話は別だけど），コマンドモードで起動するようにしておいたほうが無難だ。その方法は取扱説明書に詳しく書いてあるから知っているね。

お勧めの参考書について

僕の部屋の一角には，パソコン関連の本が山になっている。本の山というときちんと積み重ねられたものを想像するだろうけど，文字どおり山（漢字ってのは象形文字なんだぜ）になっているのだから，頭が痛い。なにか調べものがあるときには，その山をひっくり返して，必要な本を発掘するのだ。

プログラム作りにはなんらかの資料が欠かせない。BASICを使うときにはBASICのマニュアル，Cを使うときにはCのライブラリマニュアルといった具合にだ。アセンブラプログラミングをするときにも，それなりの資料を用意しなければならない。X68000上でマシン語プログラムを作るとなると，CPUであるモトローラ68000の機能や命令の働きをまとめた参考書とHuman68kのDOSコール，IOCSコールの一覧が要るだろう。ゆくゆくは，ハードウェアの解析書もほしくなるかもしれない。

●マニュアル

ところが，これらの資料は思ったよりも方々に分散されている。いちばんまとまっているのは，C compiler PRO-68KやTHE福袋V2.0に含まれる「アセンブラマニュアル」と「プログラマーズマニュアル」で，この2冊があれば，MPU68000のマシン語からDOSコール・IOCSコールの一覧までの情報がほとんど得られる。ただ，アセンブラマニュアルの68000の命令解説はアルファベット順に並んでいるので命令の関連がつかみにくく，あまり学習用には向いていない。あくまでリファレンスマニュアルだからね。また，ちょっと気持ちの悪い説明も見受けられるので，できれば，市販のまともな参考書が1冊ほしいところだ。

●MPU68000の解説書

MPU68000の参考書には，これ！というものはないが，周りの人の意見も聞いたところ「まぁ，あれだろうね」ということで一致をみた「68000プログラマーズ・ハンドブック」（2,900円，技術評論社）を勧めておく。この本は，基礎知識と簡単なプログラム例，各命令の詳細からなり，基礎知識の部分は技術者向けでちょっと難しいかもしれないが，ざっと目を通しておくとマシン語に対する理解を深めることができるだろう。

●X68000の資料

さて，68000の解説書の次は，X68000本体の資料だ。プログラマーズマニュアル以外でDOSコールやIOCSコールがまとめられているものというと，「X68000データブック」（2,900円，小学館）がある。この本は主にX68000のハードウェアの資料を並べたもので，システムコールの解説はどちらかというとオマケだが，1冊でハード・ソフト両方の資料が載っているというのはお得かもしれない。また，元祖X68000の「Human68kユーザーズマニュアル」には，付録としてHuman68kのDOSコール一覧が載っており，Oh!X 1987年7月号にはIOCSコールがまとめられている。似たような情報はほかのパソコン雑誌にも掲載されていることがあるので探してみるのもいいだろう。

このほかに，プログラム中に参照する機会が多い資料としては，Human68kユーザーズマニュアルにあるコントロールコードやエスケープシーケンスの表，取扱説明書などに見られる漢字コード表，なにかを見れば必ず載っているASCIIコード表といったものだろうか。あとは将来，ハードウェアの解説が必要になれば，「X68000テクニカルデータブック」（3,000円，アスキー）か，さっきの小学館の本がある。

あ，それから，いまのところ無視してかまわないが，FLOATn.Xを組み込むことで使えるようになる算術演算関係のシステムコール（今月の中森さんの記事参照）の解説は数値演算プロセッサーボードのマニュアルに載っている。あると便利な資料なんだけどね……。

＊　＊　＊

ツールも参考書も揃え，環境も整備したら，じっくり気を練りつつ来月を待とう。エディタの使い方には慣れておいたほうがいいな。来月は，DOSコールの呼び出し方とか，MPU68000の基本とかいった，ほんのさわり部分をやる予定だ。

ま，気楽に最後までお付き合いください。

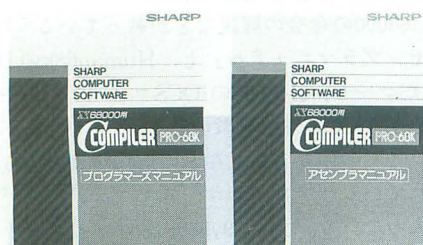
無条件ご推薦アイテム

参考文献

宍倉幸則，「68000プログラマーズ・ハンドブック」，技術評論社
Human68kユーザーズマニュアル（CZ-600C同梱），シャープ
Human68kユーザーズマニュアル（CZ-611C同梱），シャープ
BASICマニュアル（CZ-600C同梱），シャープ
BASICマニュアル（CZ-611C同梱），シャープ
プログラマーズマニュアル，シャープ
アセンブラマニュアル，シャープ

続・アセンブラによるX68000料理教室

システムコールのしくみを探ろう

X68000にはたくさんの機能を持ったIOCSコールが用意されています。しかし，その中身についてはあまり知られていないようです。X68000のシステムにより深く迫ってみましょう。実際にX68000のDOSコールとIOCSコールを作ってみました。

Nakamori Akira 中森　章

前号の特集ではDOSコールやIOCSコールをいろいろと説明してきましたが，たぶん十分には理解できなかったと思います。これらの機能を完全に理解するためにはやはり自分でROMの内容を読んで，実際に何が行われているか知るのがいちばんでしょう。人の書いたプログラムが読みにくいのは確かですが，自分のプログラミング能力を向上させるのにいい勉強になりますし，まさに一石二鳥です。

どこを逆アセンブルするか？

ROMのプログラムを読むためには，その内容を逆アセンブルするわけですが，どこのアドレスを逆アセンブルしたらよいかは問題です。ひとつの手掛かりはDOSコールのための未実装命令が実行されたときの処理，あるいはIOCSコールであるTRAP命令が実行されたときの処理を追いかけることです。

TRAP命令に関してはDB.X（デバッガ）のトレースコマンドで追っていけば大体の処理内容はわかりますが，トレースコマンドではDOSコールである未実装命令をトレースすることができません。もっと根本的な方法が必要です。そのためには，MC68000内で不当命令やTRAP命令による例外が発生したときに何が起きるのかを知っておかなければなりません（TRAP命令は意図的に例外を起こす命令）。

MC68000では，例外が発生すると，その例外の種類に応じたベクタアドレスから例外処理ルーチンのアドレスを読み込みます。そして，スタックポインタをUSPからSSPに切り換えて，例外の発生した命令のプログラムカウンタとステータスレジスタをスタックにプッシュし，読み込んだ処理アドレスにジャンプします。このとき動作モードはユーザーモードからシステムモードに切り換わります。例外が発生したときのベクタアドレスは0番地から始まっています。表1にMC68000の割り込みベクタ割り当てを示します。

表1を見るとDOSコール（ラインFエミュレータ）のベクタは2CH番地，IOCSコールのベクタ（TRAP＃15）は0BCH 番地であることがわかります。そこで，DB.Xでこれらのベクタの内容を見れば，実際の例外処理ルーチンのアドレスを知ることができるのです。例外処理ルーチンのアドレスがわかれば，そこを逆アセンブルすればよいわけですね。

表1　例外ベクタの割り当て

ベクタ番号	アドレス(16進)	割り当て
0	000	リセット直後のSSP
1	004	リセット直後のPC
2	008	バスエラー
3	00C	アドレスエラー
4	010	不当命令
5	014	ゼロ除算
6	018	CHK命令による例外
7	01C	TRAPV命令による例外
8	020	特権違反例外
9	024	トレース
10	028	ラインAエミュレータ
11	02C	ラインFエミュレータ
12	030	未使用
13	034	未使用
14	038	MC68000では未使用
15	03C	アンイニシャライズド割り込み
16-23	040-05C	未使用
24	060	スプリアス割り込み
25	064	レベル1割り込み
26	068	レベル2割り込み
27	06C	レベル3割り込み
28	070	レベル4割り込み
29	074	レベル5割り込み
30	078	レベル6割り込み
31	07C	レベル7割り込み
32-47	080-0BC	TRAP命令0～15
48-63	0C0-0FC	未使用
64-255	100-3FF	ユーザー用割り込みベクタ

新しい命令を追加する

本来なら，DOSコールやIOCSコールの処理ルーチンでどういうことが行われているのか説明したいのですが，いうまでもなくOSやBIOSのシステムサブルーチンはメーカーであるシャープやOSを作ったハドソンソフトの著作物です。公開されていない内部ルーチンを誌上で暴くのはマナーに反するのでやめておきましょう。

その代わりというか，なんとここでは新しいDOSコールと新しいIOCSコールを実際に作ることにしました。それによって，OSコールやIOCSコールというものの処理ルーチンでどういうことが行われているかだいたいの処理手順を皆さんにも感じてもらおうと思います。

新しいDOSコールとしては，X68000では使用していない（と思われる）ラインAエミュレータ（ベクタアドレス28H)を利用することにします。また，新しいIOCSコールとしてはこれまた使用していない（と思われる）TRAP命令の0番を利用することにしましょう[1)]。

ベクタの書き換えにはDOSコールの_INTVCS(0FF25H)を使用します。このDOSコールでは，引数として例外処理ルーチンのアドレスと例外ベクタ番号（ラインAエミュレータは0AH，TRAP＃0は20H)を与えます。

不当命令と例外処理

不当命令とは命令コードのうち，動作が定義されていない命令です。不当命令を実行しようとすると，CPUは「こんな命令は実行できないよ」と例外を発生し，特定のアドレスに制御が移ります。

MC68000では命令コードが0FxxxHである命令と0AxxxH である命令は未実装命令と，してほかの不当命令とは区別し，他の不当命令とは違うアドレスに制御が移るようになっています。なお，0FxxxHを実行しようとしたときの例外は「ラインF」エミュレータ，0AxxxHを実行しようとしたときの例外は「ラインA」エミュレータと呼ばれ，OSを作成した人が独自の目的で使えるようになっています。

Human68kでは「ラインF」命令の一部をDOSコールに割り当てています。また，MC68020では「ラインF」命令はコプロセッサ命令として実装されており例外は発生しません。

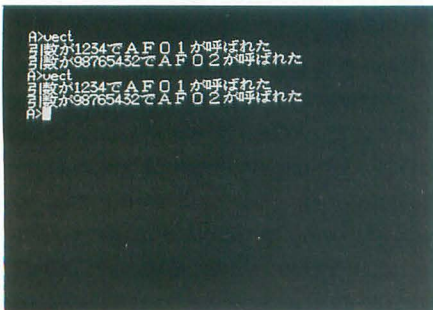



写真1

それでは，さっそくそれぞれのシステムコールについて解説していきましょう。

まずは新しいDOSコールから

最初は新しいDOSコールです。ラインA命令のうち0AF01Hと0AF02Hを有効なDOSコールとし，それ以外は従来どおりの処理をさせることにしましょう。リスト1は，この新しいDOSコールを行うためのプログラムです。ただし，ここではFloatn.xが必要ですので注意してください。

DOSコール_INTVCSで新たなラインAエミュレータの処理アドレスを設定したら，立て続けに0AF01Hと0AF02Hという新 DOSコールを実行します。DOS コールは引数をスタックを介して渡すようになっているので，0AF01Hではワードデータを，0AF02Hではロングワードデータをスタックに積んでいます。

リスト1のプログラムでは，DOSコールといってもべつに特別のことをしているわけではなく，そのDOSコールが実行されたことをメッセージで示すだけです。が，その実行部を変更することでもっといろいろな処理をすることができます。このプログラムで肝心な部分はラインA命令が呼び出されてから，それが正しいDOSコールかどうかを判定しそれぞれの処理ルーチン制御を移す部分で，次のようになっています。

1) 処理ルーチンで使うレジスタの値を退避する。
2) スタックからプログラムカウンタを取り出し，そのアドレスの1バイト目が0AFHであるか調べる。もし，0AFHでなければ従来どおりの処理をする。
3) 未実装命令の1バイト目が0AFHである場合，スタックからステータスレジスタを取り出し，その不当命令がスーパーバイザモードで実行されたのかユーザーモードで実行されたのかを調べる。それによってDOSコールの引数がSSPにあるかUSPにあるか判断し，引数の先頭アドレスをレジスタA6に入れる。
4) 未実装命令の2バイト目が01Hあるいは02Hであるか調べる。もし，そうでなければ従来どおりの処理をする。
5) 未実装命令の2バイト目が01Hである場合はその処理をする。
6) 未実装命令の2バイト目が02Hである場合はその処理をする。

ここでは，以上のような手順でDOSコールの処理の切り分けを行っていますが，未実装命令の1バイト目の比較，および2バイト目の比較をいろいろと工夫することでDOSコールにバリエーションを持たせることができるでしょう。なお，リスト1の実行結果は写真1のとおりです。

そして新しいIOCSコール

次は，新しいIOCSコールの番です。これを行うためのプログラムをリスト2に示します。リスト2のプログラムは，ファンクション番号を知るために命令コードを読むのではなく，レジスタD0の値を読むという点がリスト1と異なっています。引数はスタックを介して渡されることはありませんからステータスレジスタを読む必要もないわけです。

ところで，このIOCSコール(といっても，

リスト1　新しいDOSコール

```
==================   vect.s   ==================
   1: ******************************************
   2: *  割り込みベクタを変更して新命令を作る  *
   3: ******************************************
   4: _EXIT           equ     $ff00    ; プログラムの終了
   5: _INTVCS         equ     $ff25    ; 割り込みベクタの設定
   6: _PRINT          equ     $ff09    ; 文字列表示
   7: __LTOS          equ     $fe11    ; 値を文字列へ変換
   8:
   9: start:
  10:         pea     JOBADR
  11:         move.w  #$0a,-(sp)       ; ラインAエミュレータ
  12:         dc.w    _INTVCS
  13:         addq.l  #6,sp
  14:         move.l  d0,pre_vec       ; 前のベクタを退避
  15: new_op:
  16:         move.w  #1234,-(sp)
  17:         dc.w    $af01            ; 新命令(その1)
  18:         addq.l  #2,sp
  19:         move.l  #98765432,-(sp)
  20:         dc.w    $af02            ; 新命令(その2)
  21:         addq.l  #4,sp
  22: fine:
  23:         move.l  pre_vec,-(sp)    ; ベクタを前の値に戻す
  24:         move.w  #$0a,-(sp)
  25:         dc.w    _INTVCS
  26:         addq.l  #6,sp
  27:         dc.w    _EXIT
  28: pre_vec:
  29:         dc.l    0
  30:         .even
  31: JOBADR:
  32:         movem.l d6/a5-a6,-(sp)   ; +-------------------+
  33:         lea     14(sp),a6        ; |ステータスレジスタ | 12(sp)
  34:         movea.l (a6),a5          ; +-------------------+
  35:         cmpi.b  #$AF,(a5)+       ; |  PC 上位          | 14(sp)
  36:         beq     AF_function      ; +-------------------+
  37: undef:  movem.l (sp)+,d6/a5-a6   ; |  PC 下位          | 16(sp)
  38:         move.l  pre_vec,-(sp)    ; +-------------------+
  39:         rts                      ; スタックの状態
  40: AF_function:
  41:         moveq.l #0,d6
  42:         move.b  (a5)+,d6         ; ファンクション番号
  43:         move.l  a5,(a6)+         ; 戻りアドレスをスタックに書く
  44:         btst.b  #5,12(sp)        ; スーパーバイザーから呼ばれた
  45:         bne     From_Super       ; のでなければ
  46:         move    usp,a6           ; 引数はユーザースタック
  47: From_Super:
  48:         cmpi.b  #$01,d6          ; D6にファンクション番号
  49:         beq     do_AF01          ; ファンクション(AF01)
  50:         cmpi.b  #$02,d6
  51:         beq     do_AF02          ; ファンクション(AF02)
  52:         bra     undef            ; 未定義命令
  53: do_AF01:
  54:         movem.l d0/a0,-(sp)
  55:         lea.l   num_wrk,a0
  56:         clr.l   d0
  57:         move.w  (a6),d0          ; 引数を得る(ワード)
  58:         dc.w    __LTOS           ; 引数を文字列に変換
  59:         pea     mes00
  60:         dc.w    _PRINT           ; 引数を表示
  61:         addq.l  #4,sp
  62:         pea     mes01
  63:         dc.w    _PRINT           ; ここに来たことを表示
  64:         addq.l  #4,sp
  65:         movem.l (sp)+,d0/a0
  66:         bra     all_fine
  67: do_AF02:
  68:         movem.l d0/a0,-(sp)
  69:         lea.l   num_wrk,a0
  70:         move.l  (a6),d0          ; 引数を得る(ロング)
  71:         dc.w    __LTOS           ; 引数を文字列に変換
  72:         pea     mes00
```

```
73:         dc.w      _PRINT             ; 引数を表示
74:         addq.l    #4,sp
75:         pea       mes02
76:         dc.w      _PRINT             ; ここに来たことを表示
77:         addq.l    #4,sp
78:         movem.l   (sp)+,d0/a0
79: all_fine:
80:         movem.l   (sp)+,d6/a5-a6
81:         rte
82: mes00:
83:         dc.b      "引数が"
84: num_wrk:
85:         ds.b      10
86: mes01:
87:         dc.b      "でAFO1が呼ばれた",$d,$a,0
88: mes02
89:         dc.b      "でAFO2が呼ばれた",$d,$a,0
90:         end
```

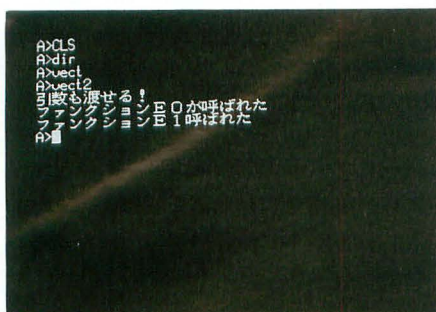



写真2

リスト2では何もしていないが）における各ファンクション番号に対する実際の処理ルーチンのアドレスは，ベクタによって与えられます。つまり，リスト2でnew_baseというラベルがベクタの先頭アドレスで，そこから4バイトごとにファンクション0の処理アドレス，ファンクション1の処理アドレス，ファンクション2の処理アドレス，……，が格納されています。

このプログラムではベクタを16個しか定義していませんから，事前にファンクションの番号が0から15の間にあることをチェックしています。もし，そうでなければ従来どおりのTRAP #0の処理を行います。また，ベクタもIOCSコールの処理が定義されているのは最初の2つのみで，それ以外は未定義として従来どおりのTRAP #0の処理を行うようになっています。

基本的には，やっている内容はリスト1のプログラムとほとんど同じですから，これまでの説明がわかっていれば特に説明はいらないと思います。写真2にリスト2のプログラムの実行結果を載せておきます。

＊　＊　＊

これまで，DOSコール，IOCSコールに始まってI/Oポートを直接操作する方法，あるいは新しいDOSコールを自分で作る方法などを示してきましたが，どうだったでしょう。誰しも自分の持っているパソコンのすべてが知りたいもの，すべての機能を扱いたいものです。

また，X68000にはそれらを満たしてくれるだけの環境が整っています。読者の皆さんも今回示した方法を参考にしてX68000の深奥に迫ってみませんか。

1）本当はTRAP #15のベクタを書き換えたかったのだが，0BCH番地に値を書き込もうとしたらX68000が暴走した。2CH 番地のほうはきっとだめだろうと思って試していない。

《参考文献》

モトローラ，「M68000マイクロプロセッサ ユーザーズ・マニュアル4th edition」，CQ出版社，1984年

POPCOM編集部・編，「X68000データブック」，小学館，1987年

リスト2　新しいIOCSコール

```
==================  vect2.s  ==================
 1: ***************************************************
 2: *  割り込みベクタを変更して新IOCSコールを作る  *
 3: ***************************************************
 4: _EXIT             equ       $ff00    ; プログラムの終了
 5: _INTVCS           equ       $ff25    ; 割り込みベクタの設定
 6: _PRINT            equ       $ff09    ; 文字列表示
 7: IOCS2   macro     number
 8:         moveq.l   #number,d0
 9:         trap      #0
10:         endm
11: start:
12:         pea       TRAP_00
13:         move.w    #$20,-(sp)          ; TRAP #00
14:         dc.w      _INTVCS
15:         addq.l    #6,sp
16:         move.l    d0,pre_vec          ; 前のベクタを退避
17: new_op:
18:         lea       mes00,a1
19:         IOCS2     $00                 ; 新IOCSコール（その1）
20:         IOCS2     $01                 ; 新IOCSコール（その2）
21: fine:
22:         move.l    pre_vec,-(sp)       ; ベクタを前の値に戻す
23:         move.w    #$0a,-(sp)
24:         dc.w      _INTVCS
25:         addq.l    #6,sp
26:         dc.w      _EXIT
27: pre_vec:
28:         dc.l      0
29:         .even
30: TRAP_00:
31:         and.l     #$ff,d0
32:         cmpi.w    #$10,d0
33:         bge       default
34: new_IOCS:
35:         move.l    a0,-(sp)
36:         lea.l     new_base,a0
37:         lsl.l     #2,d0
38:         adda.l    d0,a0               ; 新IOCSのベクタ計算
39:         movea.l   (a0),a0             ; 処理アドレスの取り出し
40:         jsr       (a0)
41:         move.l    (sp)+,a0
42:         rte
43: default:
44:         move.l    pre_vec,-(sp)       ; 新IOCSでなければ
45:         rts                           ; 本来の処理アドレスへ
46:         .even
47: new_base:
48:         dc.l      func_00             ; ファンクション00
49:         dc.l      func_01             ; ファンクション01
50:         dc.l      undef,undef         ; ファンクション02，03
51:         dc.l      undef,undef,undef,undef ; ファンクション04～
52:         dc.l      undef,undef,undef,undef
53:         dc.l      undef,undef,undef,undef
54: undef:
55:         addq.l    #4,sp               ; 戻りアドレスを捨てる
56:         movem.l   (sp)+,a0            ; $02 以上は定義していない
57:         bra       default             ; ので本来の処理アドレスへ
58: func_00:
59:         pea       (a1)
60:         dc.w      _PRINT              ; A1で示す文字列表示
61:         addq.l    #4,sp
62:         pea       mes01
63:         dc.w      _PRINT              ; ここに来たことを表示
64:         addq.l    #4,sp
65:         rts
66: func_01:
67:         pea       mes02
68:         dc.w      _PRINT              ; ここに来たことを表示
69:         addq.l    #4,sp
70:         rts
71: mes00:
72:         dc.b      "引数も渡せる！",$d,$a,0
73: mes01:
74:         dc.b      "ファンクションE0が呼ばれた",$d,$a,0
75: mes02
76:         dc.b      "ファンクションE1呼ばれた",$d,$a,0
77:         end
```

FLOAT2+.X

数値演算を高速化

Yamaguchi Tadashi
山口　正

1988年8月号では数値演算プロセッサ用のデバイスドライバを高速化したFLOAT3+.Xを掲載しましたが，今度は数値演算プロセッサなしでも利用できるFLOAT2+.Xを発表します。レイトレなど長時間の数値演算には必携のドライバです。

float2.Xも高速に

Human 68kでは数値演算をファンクションコールにまとめています。アプリケーションの多くはこのファンクションコールを使うことで実数演算，整数演算を行っており，数値演算プロセッサを使えばCで書かれたプログラムはもちろん，BASIC プログラムなどすべてが高速化の恩恵を受けることができるのです。しかし，数値演算プロセッサはまだ高価ですから，誰でもというわけにはいきません。

その数値演算プロセッサをドライブするfloat3.xもすでに長井氏によって手を加えられ，float3＋.xとして発表されています(Oh! X 1988年8月号)。float3.xではソフトの仕事はデータをプロセッサに渡すのが中心です。その部分のコードを最適化するだけで25〜30％の高速化が実現されているのですから，float2.xで演算部分まで最適化すればそれと同等以上の効果が期待できそうだと思いませんか。

まずデバッガを使ってソースを生成し，独自の改良を行ってみました。それがfloat2＋.xです。もともとfloat2.xは数値演算プロセッサMC68881の動作をエミュレートするかたちで設計されており，IEEE754に準拠したフォーマットで演算を行います。しかし，調べてみると内部ではシャープフォーマット(float1.x) とほとんど同じ処理を行い，入出力部分でだけフォーマット変換を行っているようです。それでは無駄なので内部でもIEEE 形式で演算しています。また，単精度演算は倍精度に拡張して演算していますが，これはそのままです。コンパイラを使うと単精度より倍精度のほうが速いのはこのためです。

そのほか，CLR.L D0の代わりにMOVEQ.L#0，D0を使うなど，些細な部分も1クロックでも速くなるように手を入れてあります。特に四則演算の部分は関数の中でも多用されていますので，速度を最優先にコーディングしてあります。実際，この部分の高速化がいちばん効いています。

気になる速度ですが，C-TRACE68のような実数演算を多用しているアプリケーションでは，実に約40％以上の速度向上がみこまれます。整数演算についても多少は速くなっているようです。

ラベルはCのFEFUNC.H を参考にしてつけました。__のついているところがテーブル参照で呼び出されるサブルーチンで，__Fが単精度，__Dが倍精度，__Cは値渡しにスタックを使います。IE_はIEEEフォーマットに変更した部分，FUNC_TBL：はジャンプテーブルです。

入力方法

まずリスト1のBASICプログラムを入力してください。これはX68000でキーボードから＊.x 形式のようなオブジェクトファイルを入力/作成するためのプログラムです。8ビット機のマシン語入力ツールがファイル対応になったものと考えればよいでしょう。今後もダンプリストの形態でプログラムを掲載されることがありますので，できるだけ入力しておいてください。

入力ミスがないことを確認したら，次はそのプログラムを使ってリスト2を打ち込みます。また，Cコンパイラを持っている人はコンパイルしたほうがよいでしょう。

プログラムを立ち上げると新規ファイルかどうかを聞いてきます。Yまたはそのほかのキーで選択し，エディットするファイル名を入力します。名前は適当につけてください。Eキーでエディットモードに入り，リスト2のとおりに打ち込みます。右下のCRC部分を表示するにはCコマンドを使います(なるべくコンパイルしてから)。

1987年9月号で掲載されたX-BASIC版X68000用マシン語入力ツール，または1988年6月号で掲載されたアセンブラ版マシン語入力ツールをお持ちの方はそれを使ってリスト2を打ち込んでもらっても結構です。今回の入力プログラムは1987年9月号のものをベースとしていますが，そのままでコンパイルできるように一部修正されているほか，多少なりとも使いやすいように変更を加えてみました。掲載されたままのバージョンをお使いの方はこれを参考に変更を加えてみてください。

また，キーバッファを削り取るため，やむなく未公開命令のinkey＄(0) を使用していますが，いまのところコンパイラを通しても支障はないようです。入力が終わってチェックサム，CRCチェックバイトを確認したら，ファイルサイズを整えます。

```
10 int n1,n2,d
20 char a(10186)
30 n1=fopen("new_float","r")
40 n2=fopen("float2＋.x","c")
50 d=fread(a,10186,n1)
60 d=fwrite(a,10186,n2)
70 fcloseall( )
```

上記のようなプログラムを実行すれば高速版float2＋.xのできあがりです(注：“＋”は全角です)。使用するにはエディタを使ってconfig.sysの，

device＝a:￥sys￥float2.x

などの部分を，

device＝a:￥sys￥float2＋.x

のように変更しておきます。

これと同じものがPEKIN NETにFLOAT21.Xという名前でアップロードされています(オブジェクトのみ)。通信をやっている方はそちらから入手することも可能です。詳しくはPEKIN NET (☎03(447)6182)までアクセスしてみてください。

リスト1　マシン語入力ツール

```
  10 /* program macinto-c_pro68k
  20 /* var
  30    char Dump(65535),A1
  40    int Num,Pointer=-8,Size,Data,Sum,Vsum(7)
  50    int Work(7),X,Y,F,M,CrcOn=1,EF=0
  60    str Hex,EditFile,Mode="r",Ascii,B1,Hyoji,Dam
  70 /* begin
  80 cls
  90 print "New file ( y or n )":B1=inkey$
 100 if ¬trlwr(B1)="y" then Mode="c"
 110 input "Edit file := ";EditFile
 120 Num=fopen(EditFile,Mode)
 130 Size=fseek(Num,0,2)
 140 fseek(Num,0,0)
 150 if not Size=0 then{
 160 fread(Dump,Size,Num)}
 170 fcloseall():if Size<128 then Size=128
 180 print EditFile,Size;"Byte (";hex$(Size);"H)" :print
 190 print " 'T' =P a g e  U p        'P' =P r i n t  Out"
 200 print " 'G' =P a g e  D o w n    'C' =C R C ON/OFF"
 210 print " 'E' =E d i t  M o d e    'Esc'=Command Mode"
 220 print " 'S' =S a v e            '!' =Q u i t"
 230 /*
 240 locate 0,11
 250 repeat
 260    repeat
 270      Out()
 280    until Pointer > (Size)*abs(M)
 290    M=0: B1=inkey$:B1=strlwr(B1)
 300    switch asc(B1)
 310      case  't' : Pointer=Pointer-128:break
 320      case  'g' : Pointer=Pointer+128:break
 330      case  'e' : Edit():break
 340      case  's' : Num=fopen(EditFile,"w")
 350                  fwrite(Dump,Size,Num)
 360                  fcloseall():break
 370      case  'p' : M=1:Pointer=-8:break
 380      case  'c' : CrcOn=-CrcOn:break
 390      case  '!' : EF=1
 400    endswitch
 410    Pointer=Pointer-128
 420    if Pointer<-9 then Pointer=-8
 430    locate 0,9
 440    if M=1 then print"Hit Key":B1=inkey$
 450    locate 0,9 :print"        "
 460 until EF=1
 470 end
 480 /*
 490 func Out()
 500 locate 0,11
 510 for i=0 to 7
 520   Vsum(i)=0
 530 next
 540 for i=0 to 15
 550    Pointer=Pointer+8
 560    Hex=string$(4-len(hex$(Pointer)),"0")+hex$(Pointer)
 570    Dam=inkey$(0)
 580    Pr(Hex+"  ")
 590 Ascii=""
 600    for j=0 to 7
 610      Data=Pointer+j
 620      Pr(string$(2-len(hex$(Dump(Data))),"0"))
 630      Pr(hex$(Dump(Data))+" ")
 640      Sum=Sum+Dump(Data)
 650      Vsum(j)=Vsum(j)+Dump(Data)
 660 A1=Dump(Data)
 670 if not isprint(A1) then A1=&H2E
 680      Ascii=Ascii+chr$(A1)
 690    next
 700    Pr(" : "+right$("0"+hex$(Sum),2)+"    "+Ascii)
 710    Prl()
 720    Sum=0
 730 next
 740 Pr(string$(35,"-"))
 750 Prl()
 760 Pr("SUM:  ")
 770 for i=0 to 7
 780    Pr(right$("0"+hex$(Vsum(i)),2)+" ")
 790 next
 800 if CrcOn=-1 then Pr(" "):Pr(Crc(Pointer))
 810 Prl():Prl()
 820 endfunc
 830 /* edit mode
 840 func Edit()
 850 Pointer=Pointer-120
 860 X=0:Y=0
 870 while 1
 880   for i=0 to 7
 890     Work(i)=Dump(Pointer+i)
 900   next
 910   while 1
 920   locate X/2+X+6,Y+11
 930   F=0
 940   repeat
 950     B1=inkey$:B1=strlwr(B1)
 960     switch asc(B1)
 970      case 28:X=X+1:F=1
 980             if X=16 then X=0:F=2
 990             break
1000      case 29:X=X-1:F=1
1010             if X=-1 then X=15:F=3
1020             break
1030      case 30:F=3:break
1040       case 31:F=2:break
1050       case 13:F=4:break
1060       case 27:F=5:break
1070       default
1080        if B1>="0" and B1<="9" then A1=asc(B1)-48:F=1
1090        if B1>="a" and B1<="f" then {A1=asc(B1)-87:F=1
1100                                    B1=chr$(asc(B1)-32)}
1110        if asc(B1)=12 then F=1:A1='a'-87:B1="A"
1120        if asc(B1)=47 then F=1:A1='b'-87:B1="B"
1130        if asc(B1)=42 then F=1:A1='c'-87:B1="C"
1140        if asc(B1)=45 then F=1:A1='d'-87:B1="D"
1150        if asc(B1)=43 then F=1:A1='e'-87:B1="E"
1160        if asc(B1)=61 then F=1:A1='f'-87:B1="F"
1170        if F then {Data=X/2
1180          if X and 1 then {
1190               Work(Data)=(Work(Data) and 240)+A1} else {
1200               Work(Data)=(Work(Data) mod 16)+A1*16}
1210               print B1;
1220               X=X+1
1230               if X=16 then F=4
1240               }
1250      endswitch
1260    until F
1270    if F=1 then continue
1280    if F=2 then {Y=Y+1
1290     if Y=16 then {if Pointer>65400 then {Y=15
1300       continue} else {
1310        Out()
1320        Pointer=Pointer-120
1330        if Size<Pointer then Size=Pointer+128
1340        Y=0
1350        break}} else {
1360        locate 6,Y+10
1370        for i=0 to 7
1380          print right$("0"+hex$(Dump(Pointer+i)),2);" ";
1390        next
1400        Pointer=Pointer+8}
1410       break}
1420    if F=3 then {Y=Y-1
1430     if Y=-1 then {if Pointer<118 then {Y=0
1440       continue} else {
1450        Pointer=Pointer-136
1460        Out()
1470        Y=15
1480        break}} else {
1490          locate 6,Y+12
1500          for i=0 to 7
1510            print right$("0"+hex$(Dump(Pointer+i)),2);" ";
1520          next
1530        Pointer=Pointer-8}
1540       break}
1550    if F=4 then {Sum=0
1560      for i=0 to 7
1570        Dump(Pointer+i)=Work(i)
1580        Sum=Sum+Work(i)
1590        Vsum(i)=0
1600      next
1610      locate 33,Y+11
1620      print right$("0"+hex$(Sum),2);
1630      for i=0 to 7
1640        for j=-Y to 15-Y
1650          Vsum(i)=Vsum(i)+Dump(Pointer+j*8+i)
1660        next
1670        locate 6+i*3,28
1680        print right$("0"+hex$(Vsum(i)),2)
1690      next
1700     /* CRC
1710     X=0:Y=Y+1
1720     if Y=16 then {if Pointer>65400 then Y=15 else {
1730        Out()
1740        Pointer=Pointer-120
1750        if Size<Pointer then Size=Pointer+128
1760        Y=0}} else {
1770        Pointer=Pointer+8
1780      break}}
1790    if F=5 then {Pointer=Pointer-Y*8+120
1800      break}
1810    endwhile
1820 if F=5 then Sum=0:break
1830 endwhile
1840 endfunc
1850 func Pr(St;str)
1860 if M then lprint St; else print St;
1870 endfunc
1880 func Prl()
1890 if M then lprint else print
1900 endfunc
1910 func str Crc(P)
1920 int i,j
1930 int A,C,MASK
1940 P=P-120
1950 C=Dump(P)*256+Dump(P+1)
1960 for i=2  to 127
1970    MASK=&H80:D=Dump(P+i)
1980    for j=0 to 7
1990      C=(C shl 1)
2000      if (D and MASK) then C=C+1
2010      if (C and &H10000) then C=C xor &H11021
2020      MASK=MASK shr 1
2030    next
2040 next
2050 return(string$(4-len(hex$(C)),"0")+hex$(C))
2060 endfunc
```

リスト2 FLOAT2+.X

```
0000  48 55 00 00 00 00 00 00  : 9D
0008  00 00 00 D0 00 00 28 6E  : 66
0010  00 00 00 00 00 00 00 00  : 00
0018  00 00 02 0C 00 00 00 00  : 0E
0020  00 00 00 00 00 00 00 00  : 00
0028  00 00 00 00 00 00 00 00  : 00
0030  00 00 00 00 00 00 00 00  : 00
0038  00 00 00 00 00 00 00 00  : 00
0040  FF FF FF FF 80 00 00 00  : 7C
0048  00 1A 00 00 00 22 46 4C  : CE
0050  4F 41 54 2A 2F 2D 00 00  : 6A
0058  00 00 23 CD 00 00 00 16  : 06
0060  4E 75 48 E7 80 04 2A 7A  : 1A
0068  FF EE 70 00 10 2D 00 02  : 9C
0070  61 10 1B 40 00 03 E0 48  : F7
0078  1B 40 00 04 4C DF 20 01  : AB
------------------------------------
SUM:  5F 62 4B FD 8B 62 98 95  4D9C

0080  4E 75 66 18 48 E7 40 40  : F0
0088  61 46 4C DF 02 02 67 0C  : 49
0090  2B 7C 00 00 28 6E 00 0E  : 4B
0098  42 80 4E 75 30 3C 50 03  : 44
00A0  4E 75 00 00 00 00 B0 BC  : 2F
00A8  00 00 FE 00 65 1E B0 BC  : ED
00B0  00 00 FF 00 64 16 2F 09  : B1
00B8  04 40 FE 00 E5 80 43 FA  : E4
00C0  00 E0 D3 C0 20 11 22 88  : 4E
00C8  22 5F 4E 75 70 FF 4E 75  : 76
00D0  72 0B 43 FA 00 86 70 80  : 30
00D8  4E 4F 23 C0 00 00 00 62  : E2
00E0  22 40 59 89 70 84 4E 4F  : D5
00E8  B0 BA 00 6C 66 16 22 7A  : EE
00F0  FF B2 72 0B 70 80 4E 4F  : BB
00F8  48 7A 00 26 FF 09 58 8F  : D7
------------------------------------
SUM:  69 2B 4D 81 25 00 BF 5E  0FFF

0100  42 80 4E 75 48 7A 07 74  : C2
0108  FF 09 58 8F 70 FF 4E 75  : 21
0110  61 BE 67 0A 42 67 2F 3C  : A4
0118  00 00 28 6E FF 31 FF 00  : C5
0120  0D 0A 82 B7 82 C5 82 C9  : E2
0128  95 82 93 AE 8F AC 90 94  : B7
0130  93 5F 89 89 8E 5A 83 70  : DF
0138  83 62 83 50 81 5B 83 57  : 6E
0140  82 CD 93 6F 98 5E 82 B3  : 7C
0148  82 EA 82 C4 82 A2 82 DC  : 34
0150  82 B7 0D 0A 00 00 46 45  : DB
0158  66 6E 48 E7 02 06 4D EF  : 47
0160  00 0E 2A 56 3C 1D 9C 7C  : FF
0168  FE 00 0C 46 01 00 64 26  : DB
0170  2C CD 08 2F 00 05 00 0C  : 41
0178  66 02 4E 6E E5 46 2A 7B  : F4
------------------------------------
SUM:  D6 4D 4C 17 57 A5 5C 35  E843

0180  60 20 4E 95 40 C6 1F 46  : CE
0188  00 0D 4C DF 60 40 4E 73  : 99
0190  00 7C 80 00 4E 73 4C DF  : E8
0198  60 40 2F 3A FF 06 4E 75  : D1
01A0  00 00 0A EE 00 00 0B A4  : A7
01A8  00 00 0B F8 00 00 01 5E  : 62
01B0  00 00 0B 46 00 00 0C 28  : 85
01B8  00 00 26 A6 00 00 01 5E  : 2B
01C0  00 00 0B 62 00 00 07 08  : 7C
01C8  00 00 01 5E 00 00 01 5E  : BE
01D0  00 00 26 28 00 00 26 2E  : A2
01D8  00 00 26 5E 00 00 01 5E  : E3
01E0  00 00 13 64 00 00 12 88  : 11
01E8  00 00 14 34 00 00 14 D0  : 2C
01F0  00 00 13 EC 00 00 14 A6  : B9
01F8  00 00 14 62 00 00 14 8A  : 14
------------------------------------
SUM:  C0 E9 35 AC ED 7F 9D 0F  E337

0200  00 00 12 48 00 00 01 5E  : B9
0208  00 00 0A 02 00 00 0A 6A  : 80
0210  00 00 26 D2 00 00 27 28  : 47
0218  00 00 27 A8 00 00 27 E0  : D6
0220  00 00 13 1C 00 00 18 F2  : 39
0228  00 00 15 1E 00 00 12 1A  : 5F
0230  00 00 18 8C 00 00 18 BA  : 76
0238  00 00 17 A2 00 00 01 5E  : 18
0240  00 00 0C 62 00 00 0C 7A  : F4
0248  00 00 0C AE 00 00 0D 1C  : E3
0250  00 00 0E 18 00 00 0F 30  : 65
0258  00 00 10 76 00 00 1F 4E  : F3
0260  00 00 05 88 00 00 09 42  : D8
0268  00 00 08 AE 00 00 09 0C  : CB
0270  00 00 09 8A 00 00 1F 96  : 48
0278  00 00 23 12 00 00 22 E2  : 39
------------------------------------
SUM:  00 00 2F 9C 00 00 36 CE  713F

0280  00 00 22 B4 00 00 21 AA  : A1
0288  00 00 1F EC 00 00 20 E6  : 11
0290  00 00 1E A0 00 00 1D 64  : 3F
0298  00 00 1D 7C 00 00 1D 96  : 4C
02A0  00 00 05 92 00 00 01 5E  : F6
02A8  00 00 01 5E 00 00 01 5E  : BE
02B0  00 00 01 5E 00 00 01 5E  : BE
02B8  00 00 01 5E 00 00 01 5E  : BE
02C0  00 00 01 5E 00 00 07 22  : 88
02C8  00 00 07 54 00 00 0C D8  : 3F
02D0  00 00 0D C8 00 00 10 24  : 09
02D8  00 00 05 7C 00 00 05 7C  : 02
02E0  00 00 13 16 00 00 18 E8  : 29
02E8  00 00 15 18 00 00 12 10  : 4F
02F0  00 00 18 88 00 00 18 B6  : 6E
02F8  00 00 17 98 00 00 01 5E  : 0E
------------------------------------
SUM:  00 00 F5 AC 00 00 EA A8  F62C

0300  00 00 05 B8 00 00 05 D4  : 96
0308  00 00 05 BC 00 00 05 F2  : B8
0310  00 00 06 14 00 00 06 36  : 56
0318  00 00 06 58 00 00 06 82  : E6
0320  00 00 06 6E 00 00 09 38  : B5
0328  00 00 08 A4 00 00 09 02  : B7
0330  00 00 09 80 00 00 1F C4  : 6C
0338  00 00 23 08 00 00 22 D8  : 25
0340  00 00 22 AA 00 00 21 A0  : 8D
0348  00 00 1F E2 00 00 20 DC  : FD
0350  00 00 1E 96 00 00 1D 5E  : 2F
0358  00 00 1D 72 00 00 06 98  : 2D
0360  00 00 06 AE 00 00 01 5E  : 13
0368  00 00 01 5E 00 00 01 5E  : BE
0370  00 00 01 5E 00 00 01 5E  : BE
0378  00 00 01 5E 00 00 01 5E  : BE
------------------------------------
SUM:  00 00 D5 D6 00 00 D1 3E  2CFF

0380  00 00 01 5E 00 00 07 AC  : 12
0388  00 00 07 D8 00 00 06 D8  : BD
0390  00 00 06 F4 00 00 10 1A  : 24
0398  00 00 05 7C 00 00 05 7C  : 02
03A0  00 00 01 5E 00 00 01 5E  : BE
03A8  00 00 01 5E 00 00 01 5E  : BE
03B0  00 00 01 5E 00 00 01 5E  : BE
03B8  00 00 01 5E 00 00 01 5E  : BE
03C0  00 00 01 5E 00 00 01 5E  : BE
03C8  00 00 01 5E 00 00 01 5E  : BE
03D0  00 00 01 5E 00 00 01 5E  : BE
03D8  00 00 01 5E 00 00 01 5E  : BE
03E0  00 00 01 5E 00 00 01 5E  : BE
03E8  00 00 01 5E 00 00 01 5E  : BE
03F0  00 00 01 5E 00 00 01 5E  : BE
03F8  00 00 01 5E 00 00 01 5E  : BE
------------------------------------
SUM:  00 00 1F 0E 00 00 2E 82  F0B6

0400  00 00 01 5E 00 00 01 5E  : BE
0408  00 00 01 5E 00 00 01 5E  : BE
0410  00 00 01 5E 00 00 01 5E  : BE
0418  00 00 01 5E 00 00 01 5E  : BE
0420  00 00 01 5E 00 00 01 5E  : BE
0428  00 00 01 5E 00 00 01 5E  : BE
0430  00 00 01 5E 00 00 01 5E  : BE
0438  00 00 01 5E 00 00 01 5E  : BE
0440  00 00 01 5E 00 00 01 5E  : BE
0448  00 00 01 5E 00 00 01 5E  : BE
0450  00 00 01 5E 00 00 01 5E  : BE
0458  00 00 01 5E 00 00 01 5E  : BE
0460  00 00 01 5E 00 00 01 5E  : BE
0468  00 00 01 5E 00 00 01 5E  : BE
0470  00 00 01 5E 00 00 01 5E  : BE
0478  00 00 01 5E 00 00 01 5E  : BE
------------------------------------
SUM:  00 00 10 E0 00 00 10 E0  444D

0480  00 00 01 5E 00 00 01 5E  : BE
0488  00 00 01 5E 00 00 01 5E  : BE
0490  00 00 01 5E 00 00 01 5E  : BE
0498  00 00 01 5E 00 00 01 5E  : BE
04A0  00 00 01 5E 00 00 01 5E  : BE
04A8  00 00 01 5E 00 00 01 5E  : BE
04B0  00 00 01 5E 00 00 01 5E  : BE
04B8  00 00 01 5E 00 00 01 5E  : BE
04C0  00 00 01 5E 00 00 01 5E  : BE
04C8  00 00 01 5E 00 00 01 5E  : BE
04D0  00 00 01 5E 00 00 01 5E  : BE
04D8  00 00 01 5E 00 00 01 5E  : BE
04E0  00 00 01 5E 00 00 01 5E  : BE
04E8  00 00 01 5E 00 00 01 5E  : BE
04F0  00 00 01 5E 00 00 01 5E  : BE
04F8  00 00 01 5E 00 00 01 5E  : BE
------------------------------------
SUM:  00 00 10 E0 00 00 10 E0  444D

0500  00 00 01 5E 00 00 01 5E  : BE
0508  00 00 01 5E 00 00 01 5E  : BE
0510  00 00 01 5E 00 00 01 5E  : BE
0518  00 00 01 5E 00 00 01 5E  : BE
0520  00 00 0A E2 00 00 0B 98  : 8F
0528  00 00 0B EC 00 00 0B 3A  : 3C
0530  00 00 0C 1C 00 00 26 9A  : E8
0538  00 00 09 F6 00 00 0A 5E  : 67
0540  00 00 26 CA 00 00 27 20  : 37
0548  00 00 27 8E 00 00 27 D4  : B0
0550  00 00 0C 6E 00 00 0D 10  : 97
0558  00 00 0E 0C 00 00 0F 24  : 4D
0560  00 00 10 6A 00 00 1F 42  : DB
0568  00 00 05 C8 00 00 05 E6  : B8
0570  00 00 06 08 00 00 06 2A  : 3E
0578  00 00 06 4C 00 00 06 76  : CE
------------------------------------
SUM:  00 00 B6 B0 00 00 E4 32  DF58

0580  00 00 0C 56 00 00 05 B0  : 17
0588  00 00 0C CC 00 00 06 D0  : AE
0590  00 00 0D BC 00 00 06 EC  : BB
0598  00 00 08 2C 00 00 00 66  : 9A
05A0  48 D6 00 01 4C DF 00 01  : 4B
05A8  4E 75 48 D6 00 03 4C DF  : 0F
05B0  00 03 4E 75 48 D6 00 0F  : F3
05B8  4C DF 00 0F 4E 75 61 00  : 5E
05C0  22 60 4C DF 00 02 4E 75  : 72
05C8  08 80 00 1F 44 FC 00 00  : E7
05D0  4E 75 48 E7 30 00 61 00  : 83
05D8  20 D0 61 00 04 66 24 3C  : 1B
05E0  40 F0 00 00 76 00 61 00  : 07
05E8  0A CE 4C DF 00 0C 4E 75  : D2
05F0  2F 00 20 16 48 7A FF AE  : D4
05F8  4A 80 4E 75 4A 80 67 06  : C4
------------------------------------
SUM:  3D 90 72 B4 62 97 A6 9B  A940

0600  08 40 00 1F 4A 80 4E 75  : F4
0608  48 E7 C0 00 4C D6 00 03  : 14
0610  48 7A FF 9C 48 E7 78 00  : 04
0618  61 00 21 C2 61 00 06 9C  : 47
0620  4C DF 00 1E 4E 75 48 E7  : 3B
0628  C0 00 4C D6 00 03 48 7A  : A7
0630  FF 7A 48 E7 78 00 61 00  : 81
0638  21 A6 61 00 07 20 61 00  : B0
0640  21 E0 4C DF 00 1E 4E 75  : 0D
0648  48 E7 C0 00 4C D6 00 03  : 14
0650  48 7A FF 58 48 E7 78 00  : C0
0658  61 00 21 82 61 00 07 FA  : 66
0660  61 00 21 BE 4C DF 00 1E  : 89
0668  4E 75 48 E7 C0 00 4C D6  : D4
0670  00 03 48 7A FF 36 48 E7  : 29
0678  78 00 61 00 21 62 61 00  : BD
------------------------------------
SUM:  5E 59 13 30 2D 27 E0 C2  554C

0680  08 F0 61 00 21 9C 4C DF  : 41
0688  00 1E 4E 75 48 E7 C0 00  : D0
0690  4C D6 00 03 48 7A FF 14  : FA
0698  48 E7 78 00 61 00 21 3E  : 67
06A0  61 00 0A 14 61 00 21 7A  : 7B
06A8  4C DF 00 1E 4E 75 08 80  : 94
06B0  00 1F 4A 80 4E 75 48 E7  : DB
06B8  C0 00 4C D6 00 03 48 7A  : A7
06C0  FE EA 48 E7 78 00 61 00  : F0
06C8  21 14 61 00 18 C2 61 00  : D1
06D0  21 50 4C DF 00 1E 4E 75  : 7D
06D8  48 E7 78 00 61 00 20 FE  : 26
06E0  61 00 16 FC 61 00 21 3A  : 2F
06E8  4C DF 00 1E 4E 75 61 00  : 6D
06F0  1F B8 48 E7 70 00 61 00  : D7
06F8  03 4A 24 3C 40 F0 00 00  : DD
------------------------------------
SUM:  60 DF B6 03 5F 2F F8 39  7BE9

0700  76 00 61 00 09 B2 61 00  : F3
0708  21 18 4C DF 00 0E 4E 75  : 35
0710  2F 00 20 16 48 7A FE 8A  : AF
0718  2F 01 61 00 20 CC 61 00  : DE
0720  05 F8 61 00 20 FC 4C DF  : A5
0728  00 02 4E 75 2F 00 20 16  : 2A
0730  48 7A FE 6E 2F 01 61 00  : BF
0738  20 B0 61 00 06 CC 61 00  : 64
0740  20 E0 4C DF 00 02 4E 75  : F0
0748  4A 81 67 10 48 E7 30 00  : A1
0750  61 00 05 2E 22 02 4C DF  : E3
0758  00 0C 4E 75 00 3C 00 01  : 0C
0760  4E 75 24 00 08 82 00 1F  : 90
0768  84 81 67 20 24 00 02 82  : 34
0770  7F F0 00 00 E1 9A E9 9A  : 6D
0778  04 82 00 00 03 FF 02 80  : 0A
------------------------------------
SUM:  82 12 CD 8A 6F 11 F3 04  2F7C

0780  80 0F FF FF 00 80 3F F0  : 3C
0788  00 00 4E 75 70 00 72 00  : A5
0790  74 00 4E 75 2F 02 06 82  : F0
0798  00 00 03 FF 2F 02 24 00  : 57
07A0  08 82 00 1F 84 81 67 2E  : 43
07A8  24 00 02 82 7F F0 00 00  : 17
07B0  E1 9A E9 9A 04 82 00 00  : 84
07B8  03 FF D4 9F B4 BC 00 00  : E5
07C0  08 00 64 1E E8 9A E0 9A  : 86
07C8  02 80 80 0F FF FF 80 82  : 11
07D0  4C DF 00 04 4E 75 58 8F  : D9
07D8  70 00 72 00 4C DF 00 04  : 11
07E0  4E 75 00 3C 00 01 4C DF  : 2B
07E8  00 04 4E 75 22 00 02 81  : 6C
07F0  7F FF FF FF 67 1E 02 81  : 84
07F8  7F 80 00 00 E1 99 E3 99  : F5
------------------------------------
SUM:  16 81 00 A3 74 D8 2D C9  A10A

0800  04 81 00 00 00 7F 02 80  : 86
0808  80 7F FF FF 00 80 3F 80  : 3C
0810  00 00 4E 75 42 80 4E 75  : 48
0818  2F 01 06 81 00 00 00 7F  : 36
0820  2F 01 22 00 02 81 7F FF  : 53
0828  FF FF 67 2C 02 81 7F 80  : 13
0830  00 00 E1 99 E3 99 04 81  : 7B
0838  00 00 00 7F D2 9F B2 BC  : 5E
0840  00 00 01 00 64 1C E2 99  : FC
0848  E0 99 02 80 80 7F FF FF  : F8
0850  80 81 4C DF 00 02 4E 75  : F1
0858  58 8F 42 80 4C DF 00 02  : D6
0860  4E 75 00 3C 00 01 4C DF  : 2B
0868  00 02 4E 75 20 3C 49 45  : AF
0870  45 45 22 3C 53 4F 46 54  : 24
0878  4E 75 0D 0A 95 82 93 AE  : 32
------------------------------------
SUM:  7A DB CB 0F 33 43 E0 E5  DB32

0880  8F AC 90 94 93 5F 89 89  : 63
0888  8E 5A 83 70 83 62 83 50  : 93
0890  81 5B 83 57 83 47 83 7E  : 81
0898  83 85 83 8C 81 5B 83 5E  : D4
```

▶わーい，ソーサリアンのX68000版が出るそーだ。それからイースもね。早くお店で姿を見られる日がくるといいーな。という，かなり本当っぽい噂がある。皆さん，ファルコムさんを励まそー。
田村　明広 (23) 神奈川県

```
08A0  20 66 6F 72 20 58 36 38  : 4D
08A8  30 30 30 20 76 65 72 73  : 70
08B0  69 6F 6E 20 31 2E 30 30  : 25
08B8  2E 0D 0A 81 69 82 68 82  : 9B
08C0  64 82 64 82 64 83 74 83  : AA
08C8  48 81 5B 83 7D 83 62 83  : 8C
08D0  67 81 6A 0D 0A 00 00 4D  : B6
08D8  2E 59 41 4D 41 47 55 43  : 35
08E0  48 49 00 00 2F 01 61 00  : 22
08E8  1F 00 48 7A FC D2 48 E7  : DE
08F0  30 00 61 06 4C DF 00 0C  : CE
08F8  4E 75 24 00 48 42 E8 4A  : A3
-------------------------------------
SUM:  2E 93 67 F9 35 11 0E E5  2D01

0900  02 42 07 FF 04 42 03 FF  : 92
0908  B4 7C 00 15 65 1E B4 7C  : F8
0910  00 34 65 06 4A 42 6B 24  : BA
0918  60 26 04 42 00 14 76 00  : 56
0920  44 FC 00 11 E4 B3 44 83  : AF
0928  C2 83 4E 75 26 3C 00 10  : 7A
0930  00 00 72 00 E4 AB 44 83  : C8
0938  C0 83 4E 75 70 00 72 00  : E8
0940  4E 75 2F 01 61 00 1E A2  : 14
0948  48 7A FC 74 48 E7 30 00  : 91
0950  4A 80 6A 9E 24 00 26 01  : 1D
0958  61 70 61 00 03 46 C5 40  : 80
0960  C7 41 67 0E 61 94 24 3C  : D2
0968  3F F0 00 00 76 00 61 00  : 06
0970  04 E8 4C DF 00 0C 4E 75  : E6
0978  2F 01 61 00 1E 6C 48 7A  : DD
-------------------------------------
SUM:  56 13 88 57 D6 89 E6 C3  EC11

0980  FC 3E 48 E7 30 00 4A 80  : 63
0988  6B 00 FF 68 24 00 26 01  : 1D
0990  61 38 61 00 03 0E C5 40  : 10
0998  C7 41 67 1E 24 00 48 42  : 3B
09A0  02 42 7F F0 B4 7C 43 30  : 56
09A8  64 10 61 00 FF 4E 24 3C  : 82
09B0  3F F0 00 00 76 00 61 00  : 06
09B8  03 A4 4C DF 00 0C 4E 75  : A1
09C0  2F 01 61 00 1E 24 48 7A  : 95
09C8  FB F6 48 E7 30 00 24 00  : 74
09D0  48 42 E8 4A 02 42 07 FF  : 06
09D8  B4 7C 03 FF 65 4E B4 7C  : 15
09E0  04 33 64 28 26 00 02 80  : 6B
09E8  00 0F FF FF D2 81 D1 80  : B1
09F0  53 42 B4 7C 03 FF 64 F4  : 1F
09F8  08 80 00 14 66 18 C0 BC  : 96
-------------------------------------
SUM:  BC 56 E6 23 BA 30 B1 89  2922

0A00  00 0F FF FF 4A 80 66 E4  : 21
0A08  4A 81 66 E0 70 00 72 00  : F3
0A10  4C DF 00 0C 4E 75 C0 BC  : 76
0A18  00 0F FF FF 4A 83 6A 04  : 48
0A20  08 C0 00 1F E9 4A 48 42  : A4
0A28  42 42 80 82 02 3C 00 FE  : C2
0A30  4C DF 00 0C 4E 75 48 E7  : 29
0A38  C0 00 4C D6 00 03 48 7A  : A7
0A40  FB 6A 4A 80 67 42 2F 02  : 09
0A48  2F 00 02 97 80 00 00 00  : 48
0A50  72 00 4A 80 6A 02 44 80  : 6C
0A58  24 3C 00 00 41 30 B0 BC  : 3D
0A60  00 20 00 00 64 26 60 06  : 10
0A68  04 42 00 10 D0 80 08 00  : AE
0A70  00 14 67 F4 48 40 02 40  : 39
0A78  00 0F 80 42 C0 7C 7F FF  : 8B
-------------------------------------
SUM:  B0 8A AD 4A 59 4C E6 C8  3F15

0A80  48 40 80 9F 24 1F 4E 75  : AD
0A88  72 00 4E 75 06 42 00 10  : 8D
0A90  E2 88 E2 91 B0 BC 00 20  : 69
0A98  00 00 64 F0 60 D6 48 E7  : B9
0AA0  C0 00 4C D6 00 03 48 7A  : A7
0AA8  FB 02 2F 02 24 00 48 42  : DC
0AB0  E8 4A 02 42 07 FF 04 42  : C2
0AB8  03 FF 65 3C B4 7C 00 1F  : F2
0AC0  64 3C 2F 00 48 40 80 7C  : 53
0AC8  00 10 C0 7C 00 1F 48 40  : F3
0AD0  B4 7C 00 14 64 1A E2 88  : 2C
0AD8  52 42 B4 7C 00 14 65 F6  : 33
0AE0  4A 9F 6A 02 44 80 24 1F  : 5C
0AE8  4E 75 D2 81 D1 80 53 42  : FC
0AF0  B4 7C 00 14 66 F4 60 E8  : E6
0AF8  42 80 24 1F 4E 75 24 00  : EC
-------------------------------------
SUM:  3A 2D F9 AD 8E 67 34 2C  BF4F

0B00  E2 88 E2 91 E2 88 E2 91  : BA
0B08  E2 88 E2 91 E2 88 E2 91  : BA
0B10  32 00 48 41 20 01 D4 82  : 32
0B18  E2 90 24 1F 44 FC 00 03  : F8
0B20  4E 75 48 E7 C0 00 4C D6  : D4
0B28  00 03 48 7A FA 7E 48 E7  : 6C
0B30  61 00 2E 00 67 3C 6A 02  : 9E
0B38  44 80 B3 87 4A 81 67 32  : 62
0B40  6A 02 44 81 61 64 4A 80  : C0
0B48  66 1E 20 01 4A 87 6A 14  : F4
0B50  B0 BC 80 00 00 00 62 10  : 5E
0B58  44 80 02 3C 00 FE 4C DF  : 2B
0B60  00 86 4E 75 4A 80 6A 04  : 81
0B68  44 FC 00 03 4C DF 00 86  : F4
0B70  4E 75 42 80 4C DF 00 86  : 36
0B78  4E 75 48 E7 C0 00 4C D6  : D4
-------------------------------------
SUM:  6F 60 5F 07 E0 6F 15 01  DF10
0B80  00 03 48 7A FA 26 48 E7  : 14
0B88  60 00 61 1E 4A 80 66 08  : 17
0B90  20 01 4C DF 00 06 4E 75  : 15
0B98  44 FC 00 03 4C DF 00 06  : 74
0BA0  4E 75 2F 02 61 04 24 1F  : 9C
0BA8  4E 75 34 01 C4 C0 2F 02  : AD
0BB0  42 A7 24 00 48 42 C4 C1  : 1C
0BB8  D5 AF 00 02 64 02 52 57  : 95
0BC0  48 41 24 00 C4 C1 D5 AF  : B6
0BC8  00 02 64 02 52 57 48 40  : 99
0BD0  C0 C1 D0 9F 22 1F 4E 75  : F4
0BD8  48 E7 C0 00 4C D6 00 03  : 14
0BE0  48 7A F9 C8 48 E7 71 00  : 23
0BE8  2E 00 67 22 6A 02 44 80  : E7
0BF0  B3 87 4A 81 67 2C 6A 02  : 04
0BF8  44 81 61 00 00 84 4A 87  : 7B
-------------------------------------
SUM:  34 AD 9F 8B FE 39 39 13  6AEF

0C00  6A 0C 44 80 02 3C 00 FE  : 76
0C08  4C DF 00 8E 4E 75 B0 BC  : E8
0C10  80 00 00 00 66 EE 53 80  : A7
0C18  44 FC 00 03 4C DF 00 8E  : FC
0C20  4E 75 4C DF 00 8E 00 3C  : B8
0C28  00 01 4E 75 48 E7 C0 00  : B3
0C30  4C D6 00 03 48 7A F9 74  : 54
0C38  48 E7 71 00 2E 00 67 CE  : 03
0C40  6A 02 44 80 4A 81 67 DA  : 3C
0C48  6A 02 44 81 61 32 4A 87  : 95
0C50  6A 02 44 82 20 02 4C DF  : 7F
0C58  00 8E 4E 75 48 E7 C0 00  : 40
0C60  4C D6 00 03 48 7A F9 44  : 24
0C68  48 E7 71 00 4A 80 67 0A  : DB
0C70  4A 81 67 AE 61 0A 02 3C  : 89
0C78  00 FE 4C DF 00 8E 4E 75  : 7A
-------------------------------------
SUM:  78 EA 8D F0 C6 9B 90 85  0851

0C80  74 00 76 1F D0 80 D5 82  : B0
0C88  B4 81 65 04 52 40 94 81  : 45
0C90  51 CB FF F2 4E 75 48 E7  : FF
0C98  C0 00 4C D6 00 03 48 7A  : A7
0CA0  F9 0E 4A 80 66 06 4A 81  : 08
0CA8  02 3C 00 04 4E 75 48 E7  : 34
0CB0  F0 00 4C D6 00 0F 48 7A  : E3
0CB8  F9 00 61 0C 02 3C 00 F5  : 99
0CC0  64 04 00 3C 00 08 4E 75  : 6F
0CC8  2F 00 B5 9F 6B 0E 4A 80  : C6
0CD0  6A 14 61 12 67 F0 0A 3C  : 8E
0CD8  00 01 4E 75 E3 98 E2 98  : B9
0CE0  02 3C 00 FB 4E 75 B0 82  : 2E
0CE8  66 02 B2 83 4E 75 48 7A  : 22
0CF0  FF B2 4A 81 67 06 08 40  : 31
0CF8  00 1F 4E 75 2F 00 02 9F  : B2
-------------------------------------
SUM:  81 BE CB 27 0D 8C 59 DF  C93E

0D00  7F FF FF FF 67 04 08 40  : 2F
0D08  00 1F 4E 75 48 E7 C0 00  : D1
0D10  4C D6 00 03 48 7A F8 94  : 73
0D18  48 E7 3E 00 24 3C 3F F0  : FC
0D20  00 00 76 00 61 3E 44 C6  : 1F
0D28  4C DF 00 7C 4E 75 4A 80  : 34
0D30  6A 00 01 3C C1 42 C3 43  : B0
0D38  60 00 01 34 4A 81 66 3C  : 02
0D40  20 02 22 03 08 80 00 1F  : EE
0D48  4E 75 4A 83 66 36 4E 75  : EF
0D50  48 E7 F0 00 4C D6 00 0F  : 50
0D58  48 7A F8 5A 48 E7 3E 00  : 81
0D60  48 7A FF C4 28 00 B5 84  : E6
0D68  6B C4 4A 80 6A 04 48 7A  : 29
0D70  FF 82 7C 00 02 80 7F FF  : FD
0D78  FF FF 67 C0 02 82 7F FF  : 27
-------------------------------------
SUM:  D8 51 83 47 6D 90 3D 28  1B55

0D80  FF FF 67 C6 28 00 48 44  : DF
0D88  E8 44 2A 02 48 45 E8 45  : 12
0D90  B8 45 64 06 C1 42 C3 43  : 70
0D98  C9 45 98 45 DA 44 B8 7C  : 3D
0DA0  00 35 64 56 48 40 48 42  : 01
0DA8  80 7C 00 10 84 7C 00 10  : 1C
0DB0  C0 7C 00 1F C4 7C 00 1F  : BA
0DB8  48 40 48 42 B8 7C 00 20  : 66
0DC0  65 14 D6 83 40 E7 26 02  : 21
0DC8  74 00 44 DF 02 44 00 1F  : FC
0DD0  60 04 E2 8A E2 93 51 CC  : 62
0DD8  FF FA D3 83 D1 82 08 00  : AA
0DE0  00 15 67 06 E2 88 E2 91  : 5F
0DE8  52 45 E9 4D 6B 00 01 42  : 7B
0DF0  48 40 02 40 00 0F 80 45  : 9E
0DF8  48 40 4E 75 48 E7 C0 00  : 3A
-------------------------------------
SUM:  0A 26 A8 51 DD 3D 95 DE  CA2E

0E00  4C D6 00 03 48 7A F7 A4  : 82
0E08  48 E7 3E 00 24 3C 3F F0  : FC
0E10  00 00 76 00 61 4A 44 C6  : 2B
0E18  4C DF 00 7C 4E 75 4A 80  : 34
0E20  6A 00 FF 50 61 00 FF 4C  : 65
0E28  60 00 FE C8 4A 81 66 48  : 9F
0E30  20 02 22 03 08 80 00 1F  : EE
0E38  60 00 FE B8 4A 83 66 40  : 89
0E40  4E 75 B2 83 66 3E 70 00  : 0C
0E48  72 00 4E 75 48 E7 F0 00  : 54
0E50  4C D6 00 0F 48 7A F7 5E  : 48
0E58  48 E7 3E 00 48 7A FF B8  : E6
0E60  28 00 B5 84 6B B8 4A 80  : 4E
0E68  6A 04 C1 42 C3 43 7C 00  : F3
0E70  02 80 7F FF FF FF 67 B4  : 19
0E78  02 82 7F FF FF FF 67 BC  : 23
-------------------------------------
SUM:  14 D6 83 1D 82 0B 79 D3  FD4E

0E80  B0 82 67 BE 64 08 C1 42  : C6
0E88  C3 43 48 7A FE 66 28 00  : 54
0E90  48 44 E8 44 2A 02 48 45  : 71
0E98  E8 45 98 45 DA 44 B8 7C  : 5C
0EA0  00 35 64 A6 48 40 48 42  : 51
0EA8  80 7C 00 10 84 7C 00 10  : 1C
0EB0  C0 7C 00 1F C4 7C 00 1F  : BA
0EB8  48 40 48 42 2F 07 7E 00  : C6
0EC0  B8 7C 00 20 65 14 2E 03  : FE
0EC8  48 47 26 02 74 00 02 44  : 71
0ED0  00 1F 60 06 E2 8A E2 93  : 66
0ED8  E2 57 51 CC FF F8 CE 7C  : 97
0EE0  FF E0 44 47 93 83 91 82  : 93
0EE8  60 08 53 45 DE 47 D3 81  : 79
0EF0  D1 80 08 00 00 14 67 F2  : C6
0EF8  74 00 DE 47 64 10 D3 82  : 62
-------------------------------------
SUM:  B1 5C 2F 9F B4 77 2D 41  D728

0F00  D1 82 08 00 00 15 67 06  : DD
0F08  E2 88 E2 91 52 45 2E 1F  : C1
0F10  BA 7C 08 00 64 0E E9 4D  : E6
0F18  48 40 C0 7C 00 0F 80 45  : 98
0F20  48 40 4E 75 4A 45 6A 08  : 4C
0F28  70 00 72 00 7C 01 4E 75  : 22
0F30  20 3C 7F FF FF FF 72 FF  : 49
0F38  7C 03 4E 75 44 C6 4C DF  : 77
0F40  01 FC 4E 75 4A 83 66 4A  : 3D
0F48  70 00 72 00 7C 00 4E 75  : 21
0F50  4A 81 66 36 7C 00 4E 75  : A6
0F58  4A 82 66 58 4A 83 66 54  : 11
0F60  60 00 02 12 48 E7 F0 00  : 93
0F68  4C D6 00 0F 48 7A F6 46  : 2F
0F70  48 E7 3F 80 48 7A FF C6  : 75
0F78  28 00 B5 84 6A 04 48 7A  : 91
-------------------------------------
SUM:  2A 01 C1 1E 8D 67 09 20  2F34

0F80  FD 72 02 80 7F FF FF FF  : 6D
0F88  67 C6 02 82 7F FF FF FF  : 2D
0F90  67 B2 48 40 48 42 3E 00  : 69
0F98  3C 02 E8 4F E8 4E 04 47  : F6
0FA0  07 FE DE 46 C0 7C 00 0F  : 74
0FA8  80 7C 00 10 48 40 C4 7C  : D4
0FB0  00 0F 67 A4 84 7C 00 10  : 2A
0FB8  48 42 4E 56 FF E8 61 42  : B8
0FC0  20 2E FF F2 22 2E FF F6  : 84
0FC8  4E 5E 74 00 E2 88 E2 91  : FD
0FD0  E2 88 E2 91 E2 88 E2 91  : BA
0FD8  08 00 00 16 67 0A 54 81  : 64
0FE0  D1 82 E2 88 E2 91 60 08  : 98
0FE8  52 81 D1 82 E2 88 E2 91  : 03
0FF0  08 00 00 15 67 00 01 7E  : 03
0FF8  E2 88 E2 91 52 47 60 00  : D6
-------------------------------------
SUM:  3B 56 B1 2A 83 56 1F D2  02FD

1000  01 74 78 00 20 4E 21 04  : 80
1008  21 04 21 04 48 E0 38 00  : AA
1010  4A 81 67 10 41 EE FF F6  : 66
1018  C1 41 61 12 55 48 48 40  : 9A
1020  61 0C C1 41 41 EE FF F2  : 8F
1028  61 04 55 48 48 40 4C AE  : 84
1030  00 3C FF E8 CA C0 C6 C0  : 33
1038  C8 C0 C4 C0 DB A8 00 06  : 95
1040  2C 28 00 02 DD 83 64 02  : 1C
1048  52 50 21 46 00 02 D9 A8  : 8C
1050  00 04 2C 10 DD 82 20 86  : 45
1058  4E 75 2F 01 61 00 17 8A  : F5
1060  48 7A F5 5C 61 00 FC 3C  : AC
1068  67 12 2F 00 02 80 7F F0  : 99
1070  00 00 67 0A 20 1F 04 80  : 34
1078  00 10 00 00 4E 75 20 1F  : 12
-------------------------------------
SUM:  32 D3 41 16 18 15 C4 25  767F

1080  44 FC 00 01 4E 75 44 C6  : 0E
1088  4C DF 00 FC 4E 75 4A 81  : B5
1090  66 46 42 86 4E 75 4A 83  : 04
1098  66 36 60 00 01 0E 4A 82  : D7
10A0  66 54 4A 83 66 50 60 00  : 9D
10A8  00 CC 48 E7 F0 00 4C D6  : 0D
10B0  00 0F 48 7A F5 00 48 E7  : F5
10B8  3F 00 48 7A FF CA 28 00  : F2
10C0  B5 84 6A 04 48 7A FC 2C  : 91
10C8  02 82 7F FF FF FF 67 C6  : 2D
10D0  02 80 7F FF FF FF 67 B6  : 1B
10D8  48 40 48 42 3E 00 3C 02  : 8E
10E0  E8 4F E8 4E 9E 46 C0 7C  : 8D
10E8  00 0F 80 7C 00 10 48 40  : A3
10F0  C4 7C 00 0F 67 A8 84 7C  : 5E
10F8  00 10 48 42 78 00 B0 82  : 44
-------------------------------------
SUM:  AE 36 24 40 36 FD 80 6D  06DC

1100  66 02 B2 83 64 06 53 47  : A1
1108  7A 00 60 06 7A 01 92 83  : 70
1110  91 82 DA 85 65 1E D2 81  : 48
1118  D1 80 B0 82 65 F4 66 04  : 46
1120  B2 83 65 EE 92 83 91 82  : B0
1128  52 45 60 E6 08 04 00 14  : FD
1130  66 1E DA 85 D9 84 D2 81  : 93
1138  D1 80 B0 82 65 EE 66 04  : 40
1140  B2 83 65 E8 92 83 91 82  : AA
1148  52 45 08 04 00 14 67 E2  : 00
1150  D2 81 D1 80 B0 82 65 18  : 53
1158  66 04 B2 83 65 12 70 00  : 86
1160  52 85 D9 80 08 04 00 15  : 51
1168  67 06 E2 8C E2 95 52 47  : EB
1170  20 04 22 05 7C 00 06 47  : 14
1178  03 FF 67 1E BE 7C 08 00  : C9
-------------------------------------
SUM:  95 45 1F 89 4B 52 13 89  6161

1180  64 0E 48 40 02 40 00 0F  : 4B
1188  E9 4F 80 47 48 40 4E 75  : 4A
```

▶まっ，アレですね。缶詰のレビューを書きそこなった僕としてはですね，なんかスゲー恥をさらしてしまったような気がするんですけど，それはほっぽっといて，こういいたいわけですよ，「誰かTOTOをOh!X LIVEでやろーぜ。TOTOROでもいーいでよ」。

田中　真一（18）三重県

```
1190  4A 47 6B 00 FD 94 60 00  : ED
1198  FD 98 02 84 00 0F FF FF  : 28
11A0  66 E0 4A 85 66 DC 60 00  : B7
11A8  FD 80 20 3C 7F FF FF FF  : 55
11B0  72 FF 7C 05 4E 75 61 12  : 28
11B8  64 0E B0 3C 00 41 65 08  : 0C
11C0  B0 3C 00 5B 0A 3C 00 01  : 8E
11C8  4E 75 B0 3C 00 61 65 08  : 7D
11D0  B0 3C 00 7B 0A 3C 00 01  : AE
11D8  4E 75 B0 3C 00 30 65 08  : 4C
11E0  B0 3C 00 3A 0A 3C 00 01  : 6D
11E8  4E 75 B0 3C 00 30 65 08  : 4C
11F0  B0 3C 00 32 0A 3C 00 01  : 65
11F8  4E 75 B0 3C 00 30 65 08  : 4C
-----------------------------------
SUM:  C5 6D 8B 3F A2 95 66 C0  4831

1200  B0 3C 00 38 0A 3C 00 01  : 6B
1208  4E 75 61 CE 64 1A B0 3C  : 5C
1210  00 41 65 14 B0 3C 00 47  : ED
1218  65 0A B0 3C 00 61 65 08  : 29
1220  B0 3C 00 67 0A 3C 00 01  : 9A
1228  4E 75 61 9E 65 04 90 3C  : F7
1230  00 20 4E 75 61 84 65 04  : 31
1238  D0 3C 00 20 4E 75 61 F4  : 44
1240  B0 3C 00 61 65 04 D0 3C  : C2
1248  FF D9 D0 3C FF D0 4E 75  : 76
1250  2F 01 61 00 15 94 48 7A  : FC
1258  F3 66 74 0E 2F 08 61 00  : 73
1260  05 82 20 57 61 06 20 5F  : E4
1268  60 00 06 4C 10 18 67 16  : 57
1270  B0 3C 00 2E 66 F6 10 10  : 96
1278  61 00 FF 60 64 08 53 88  : 07
-----------------------------------
SUM:  78 43 EF CC 1F B8 1C F9  2F0A

1280  10 E8 00 01 66 FA 4E 75  : 1C
1288  48 E7 E0 40 4F EF FF F0  : 7C
1290  22 48 20 4F 61 32 20 4F  : DB
1298  61 22 74 20 92 00 6B 08  : 1C
12A0  67 06 12 C2 53 01 60 F6  : EB
12A8  20 4F 12 D8 66 FC 53 89  : 97
12B0  20 49 4F EF 00 10 4C DF  : E2
12B8  02 07 4E 75 70 FF 52 40  : CD
12C0  4A 18 66 FA 53 88 4E 75  : 60
12C8  48 E7 E1 60 4E 56 FF F6  : 09
12D0  22 4F 45 FA 00 56 42 47  : 8F
12D8  4A 80 6A 04 44 80 46 47  : 89
12E0  24 12 67 12 42 01 52 01  : 45
12E8  90 82 64 FA D0 82 53 01  : 16
12F0  12 C1 58 8A 60 EA 42 01  : 42
12F8  22 4F 4A 11 66 0A 52 89  : 17
-----------------------------------
SUM:  6A 50 98 AD 8E 52 37 DF  6F81

1300  52 01 B2 3C 00 09 65 F2  : A1
1308  4A 47 67 04 10 FC 00 2D  : 35
1310  10 19 D0 3C 00 30 10 C0  : 35
1318  52 01 B2 3C 00 0A 65 F0  : A0
1320  42 10 4E 5E 4C DF 06 87  : B6
1328  4E 75 3B 9A CA 00 05 F5  : 5C
1330  E1 00 00 98 96 80 00 0F  : 9E
1338  42 40 00 01 86 A0 00 00  : A9
1340  27 10 00 00 03 E8 00 00  : 22
1348  00 64 00 00 00 0A 00 00  : 6E
1350  00 01 00 00 00 00 2F 01  : 31
1358  48 7A F2 64 61 00 04 32  : AF
1360  B0 3C 00 26 66 00 01 F8  : 71
1368  52 88 10 18 61 00 FE BC  : 1D
1370  B0 3C 00 42 67 12 B0 3C  : 93
1378  00 4F 67 12 B0 3C 00 48  : FC
-----------------------------------
SUM:  D2 65 8D 3F 84 7E C7 C5  6D89

1380  67 12 44 FC 00 09 4E 75  : 85
1388  61 00 01 18 60 0A 61 00  : 45
1390  00 9C 60 04 61 00 00 DE  : 3F
1398  65 08 61 00 F6 A6 44 FC  : AA
13A0  00 00 4E 75 48 E7 7F 00  : 71
13A8  4E 56 00 00 42 80 61 00  : C7
13B0  03 E0 67 44 61 00 03 C4  : B6
13B8  10 10 61 00 FE 1E 65 38  : 3A
13C0  90 3C 00 30 72 00 12 00  : 80
13C8  52 88 10 10 61 00 FE 0C  : 65
13D0  65 32 90 3C 00 30 2F 01  : C3
13D8  D2 81 65 10 D2 81 65 0C  : 8C
13E0  D2 9F 65 08 D2 81 65 04  : 9A
13E8  D2 80 64 DC 4E 5E 4C DF  : 69
13F0  00 FE 44 FC 00 03 4E 75  : 04
13F8  4E 5E 4C DF 00 FE 44 FC  : 15
-----------------------------------
SUM:  99 EE 7A 1C 65 CF 22 B8  6839

1400  00 09 4E 75 4A 47 6B 12  : DA
1408  4A 81 6B E0 20 01 4E 5E  : E3
1410  4C DF 00 FE 44 FC 00 00  : 69
1418  4E 75 4A 81 6A 04 B2 BC  : 70
1420  80 00 00 00 67 E6 60 C4  : F1
1428  44 81 60 E0 2F 01 70 00  : A5
1430  10 10 61 00 FD C6 65 34  : DD
1438  90 3C 00 30 22 00 52 88  : F8
1440  10 10 61 00 FD B6 65 12  : AB
1448  B2 BC 20 00 00 00 64 14  : 06
1450  E7 89 90 3C 00 30 82 00  : EE
1458  60 E4 20 01 22 1F 44 FC  : E6
1460  00 00 4E 75 22 1F 44 FC  : 44
1468  00 03 4E 75 22 1F 44 FC  : 47
1470  00 09 4E 75 2F 01 70 00  : 6C
1478  10 10 61 00 FD 8E 65 EC  : 5D
-----------------------------------
SUM:  61 00 40 80 5C CD DE B2  D411

1480  61 00 FD BC 22 00 52 88  : 16
1488  10 10 61 00 FD 7E 65 CA  : 2B
1490  B2 BC 10 00 00 00 64 CC  : AE
1498  E9 89 61 00 FD A2 82 00  : F4
14A0  60 E4 2F 01 70 00 10 10  : 04
14A8  61 00 FD 40 65 BE 90 3C  : 8D
14B0  00 30 22 00 52 88 10 10  : 4C
14B8  61 00 FD 30 65 9C D2 81  : E2
14C0  65 A2 90 3C 00 30 82 00  : 85
14C8  60 EA 48 E7 E0 40 4E 56  : 3D
14D0  FF DE 22 4F 72 1F 74 30  : 83
14D8  D0 80 64 02 52 02 12 C2  : DE
14E0  51 C9 FF F4 60 50 48 E7  : EC
14E8  E0 40 4E 56 FF DE 22 4F  : 12
14F0  74 0A E5 98 32 00 02 41  : 70
14F8  00 03 60 04 E7 98 32 00  : 18
-----------------------------------
SUM:  67 69 0A 87 C4 59 13 BA  0305

1500  02 41 00 07 D2 3C 00 30  : 88
1508  12 C1 51 CA FF F0 60 26  : 63
1510  48 E7 E0 40 4E 56 FF DE  : D0
1518  22 4F 74 07 E9 98 32 00  : 9F
1520  02 41 00 0F B2 3C 00 0A  : 4A
1528  65 02 5E 01 D2 3C 00 30  : 04
1530  12 C1 51 CA FF E8 42 11  : 28
1538  22 4F 0C 11 00 30 66 0A  : 2E
1540  4A 29 00 01 67 04 52 89  : BA
1548  60 F0 10 D9 66 FC 53 88  : 76
1550  4E 5E 4C DF 02 07 4E 75  : A3
1558  2F 01 48 7A F0 62 48 E7  : 73
1560  3F 60 4E 56 FF E2 61 00  : 85
1568  02 28 67 00 00 D8 61 00  : CA
1570  02 0A 3D 7C FF FF FF FA  : BC
1578  42 AE FF FC 10 10 7C 00  : 87
-----------------------------------
SUM:  C5 43 F5 04 58 DC B1 F0  9C19

1580  B0 3C 00 2E 66 0C 08 C6  : 5A
1588  00 1F 42 6E FF FA 52 88  : A2
1590  10 10 61 00 FC 46 65 00  : 28
1598  00 AC 52 88 61 00 00 CE  : B5
15A0  43 EE FF E2 12 FC 00 30  : 50
15A8  24 49 12 C0 7A 01 10 10  : DA
15B0  B0 3C 00 2E 66 12 4A 86  : 62
15B8  66 00 00 8A 08 C6 00 1F  : DD
15C0  42 6E FF FA 52 88 60 E6  : C9
15C8  61 00 FC 10 65 14 12 C0  : B8
15D0  52 88 61 00 00 98 52 45  : 6A
15D8  BA 7C 00 0F 66 D0 61 00  : DC
15E0  01 56 61 00 00 E6 61 00  : FF
15E8  00 8C 4A 47 6A 04 61 00  : EC
15F0  F7 02 2D 40 FF F2 2D 41  : C5
15F8  FF F6 4A 6E FF FA 67 24  : 31
-----------------------------------
SUM:  E3 D6 84 8C 41 FB 94 51  10F3

1600  61 00 F4 A8 65 1A 28 00  : A4
1608  61 00 F4 38 24 2E FF F2  : D0
1610  26 2E FF F6 61 00 F6 A4  : 44
1618  66 06 2D 44 FF FC 60 04  : 3C
1620  42 6E FF FA 20 2E FF F2  : E8
1628  22 2E FF F6 74 00 34 2E  : 1B
1630  FF FA 26 2E FF FC 4E 5E  : F4
1638  50 8F 4C DF 06 F0 44 FC  : 40
1640  00 00 4E 75 4E 5E 4C DF  : 9A
1648  06 FC 70 00 72 00 44 FC  : 24
1650  00 09 4E 75 4E 5E 4C DF  : A3
1658  06 FC 44 FC 00 01 4E 75  : 06
1660  4E 5E 4C DF 06 FC 44 FC  : 19
1668  00 03 4E 75 4A 86 6A 02  : 02
1670  53 46 4E 75 70 00 72 00  : 3E
1678  24 3C 40 24 00 00 76 00  : 3A
-----------------------------------
SUM:  D2 3D FC EA 50 9D 02 41  CE83

1680  61 00 F8 EE 24 00 26 01  : 92
1688  70 00 10 1A 90 3C 00 30  : 96
1690  61 00 F3 B0 61 00 F6 C6  : 21
1698  53 45 66 DC D8 46 67 28  : 87
16A0  6B 14 24 3C 40 24 00 00  : 43
16A8  76 00 61 00 F8 C4 65 B0  : A8
16B0  53 44 66 EE 4E 75 24 3C  : 0E
16B8  40 24 00 00 76 00 61 00  : 3B
16C0  F9 F6 65 90 52 44 66 EE  : CE
16C8  4E 75 78 00 B0 3C 00 45  : 6C
16D0  67 08 B0 3C 00 65 67 02  : 29
16D8  4E 75 42 6E FF FA 52 88  : 46
16E0  10 10 B0 3C 00 2B 67 0A  : A8
16E8  B0 3C 00 2D 66 08 08 C4  : 53
16F0  00 1F 52 88 10 10 61 00  : 7A
16F8  FA E2 65 00 FF 48 90 3C  : 54
-----------------------------------
SUM:  AF F6 82 E9 5F 49 EC D2  F52D

1700  00 30 18 00 42 40 52 88  : A4
1708  10 10 61 00 FA CE 65 1E  : CC
1710  90 3C 00 30 3F 04 E5 4C  : 70
1718  D8 5F D8 44 D8 40 B8 7C  : 9F
1720  03 E8 65 E2 4A 84 6A 00  : 6A
1728  FF 38 60 00 FF 28 4A 84  : 8C
1730  6A 02 44 44 4E 75 52 46  : 4F
1738  53 45 0C 21 00 35 65 0E  : 6D
1740  52 21 0C 11 00 3A 65 06  : 35
1748  12 BC 00 30 60 F2 B3 CA  : CD
1750  64 04 24 49 52 46 10 10  : 8D
1758  61 00 FA 80 64 10 B0 3C  : 3B
1760  00 2E 66 14 4A 86 6B 00  : E3
1768  FE DC 08 C6 00 1F 52 88  : A1
1770  4A 86 6B E2 52 46 60 DE  : F3
1778  4E 75 42 47 B0 3C 00 2B  : 63
-----------------------------------
SUM:  F6 28 AB C8 4C 51 B4 F3  AA37

1780  67 08 B0 3C 00 2D 66 04  : F2
1788  46 47 52 88 4E 75 52 88  : 04
1790  10 10 B0 3C 00 09 67 F6  : 72
1798  B0 3C 00 20 67 F0 4A 00  : AD
17A0  4E 75 44 80 02 42 00 FF  : CA
17A8  B0 42 63 02 30 02 61 00  : EA
17B0  00 F0 10 FC 00 30 61 00  : 8D
17B8  00 E8 10 FC 00 2E 4A 40  : AC
17C0  67 0C 61 00 00 DC 10 BC  : 7C
17C8  00 30 53 40 66 F4 61 00  : 7E
17D0  00 E6 4C DF 00 06 4E 75  : DA
17D8  2F 01 61 00 10 0C 48 7A  : 6F
17E0  ED E2 48 E7 60 00 61 00  : BF
17E8  00 E4 4A 81 67 08 61 00  : 7F
17F0  00 B0 10 FC 00 2D 4A 80  : B3
17F8  67 5A 6B 20 02 42 00 FF  : 8F
-----------------------------------
SUM:  55 1D E7 3D 26 96 88 EB  7354

1800  B0 42 67 60 64 36 41 F0  : 84
1808  08 00 61 00 00 94 10 BC  : C9
1810  00 2E 61 00 00 A2 61 00  : 92
1818  00 A4 60 48 2F 08 61 00  : E4
1820  00 96 61 00 00 98 91 D7  : F7
1828  52 88 22 00 44 81 D2 88  : 1B
1830  20 5F 02 42 00 FF B2 42  : B6
1838  65 00 FF 68 52 88 61 60  : 67
1840  10 BC 00 2E 61 70 61 74  : A0
1848  10 FC 00 45 53 80 61 1C  : A1
1850  42 10 60 10 61 4A 10 FC  : 79
1858  00 30 61 44 10 FC 00 2E  : 0F
1860  61 54 61 58 61 50 4C DF  : 4A
1868  00 06 4E 75 12 3C 00 2B  : 42
1870  4A 80 6A 06 12 3C 00 2D  : B5
1878  44 80 10 C1 32 3C 00 64  : 67
-----------------------------------
SUM:  E0 E3 F7 AD 05 4E A7 02  B661

1880  61 0E 32 3C 00 0A 61 08  : 50
1888  D0 3C 00 30 10 C0 4E 75  : CF
1890  14 3C 00 2F 52 02 90 41  : A4
1898  64 FA D0 41 10 C2 4E 75  : 04
18A0  48 E7 C0 80 12 18 10 01  : AA
18A8  12 10 10 C0 66 F8 4C DF  : 7B
18B0  01 03 4E 75 52 88 4A 10  : FB
18B8  66 FA 4E 75 0C 20 00 30  : 7F
18C0  67 FA 52 88 42 10 4E 75  : 50
18C8  61 00 0F 1E 48 E7 3F E0  : DC
18D0  4E 56 FF EA 61 00 02 FE  : EE
18D8  30 2E FF EE 61 00 03 26  : D5
18E0  30 2E FF EE 61 00 03 3A  : E9
18E8  4E 5E 30 07 48 C0 22 06  : 13
18F0  4C DF 07 FC 4E 75 61 00  : 52
18F8  0E F0 48 E7 3F E0 4E 56  : F0
-----------------------------------
SUM:  88 4D 4B 5C CA 52 99 62  3A9B

1900  FF EA 61 00 02 D0 30 2E  : 7A
1908  FF EE D0 47 61 00 02 F6  : 5D
1910  30 2E FF EE D0 47 61 00  : C3
1918  03 08 4E 5E 30 07 48 C0  : F6
1920  22 06 4C DF 07 FC 4E 75  : 19
1928  2F 01 61 00 0E BC 48 7A  : 1D
1930  EC 8E 48 E7 3F E0 4A 83  : 95
1938  6A 02 42 83 61 00 F3 64  : E9
1940  67 06 08 04 00 03 66 76  : 58
1948  08 04 00 04 67 02 52 82  : 4D
1950  48 E7 30 00 4E 56 FF EA  : EC
1958  24 03 61 00 02 78 30 2E  : 60
1960  FF EE D0 47 61 00 02 9E  : 05
1968  30 2E FF EE D0 47 61 00  : C3
1970  02 B0 61 00 01 36 61 00  : AB
1978  01 56 61 00 01 6A 61 00  : 84
-----------------------------------
SUM:  E5 BB DF 19 02 70 BA 68  BD5F

1980  01 9E 61 1C 61 00 02 10  : 8F
1988  61 00 01 BA 61 00 02 20  : 9F
1990  4E 5E 30 07 48 C0 22 06  : 13
1998  50 8F 4C DF 07 FC 4E 75  : D0
19A0  08 2E 00 03 00 17 67 14  : CB
19A8  20 6E FF EA 61 00 FF 08  : DF
19B0  10 FC 00 45 42 80 61 00  : 74
19B8  FE B4 42 10 4E 75 08 04  : D3
19C0  00 04 66 1A 08 04 00 05  : 95
19C8  66 14 08 04 00 06 66 0E  : 00
19D0  4A 82 67 0A 53 82 66 06  : 7E
19D8  4A 83 66 02 74 01 48 E7  : D9
19E0  30 00 4E 56 FF EA D4 83  : 14
19E8  61 00 01 EA 30 2E FF EE  : 97
19F0  61 00 02 12 30 2E FF EE  : C0
19F8  61 00 02 26 61 28 20 6E  : A0
-----------------------------------
SUM:  83 F4 AD A0 91 C3 49 98  3F78

1A00  FF EA 61 00 FE B2 10 FC  : 06
1A08  00 45 30 07 48 C0 61 00  : E5
1A10  FE 5C 42 10 61 2E 4E 5E  : E7
1A18  30 07 48 C0 22 06 50 8F  : 46
1A20  4C DF 07 FC 4E 75 20 2E  : 3F
1A28  00 04 4A AE 00 10 6B 10  : 87
1A30  20 6E FF EA 41 F0 00 00  : A8
1A38  61 00 FE 66 10 BC 00 2E  : BF
1A40  9E 40 4E 75 20 6E FF EA  : 18
1A48  20 2E 00 14 08 00 00 04  : 6E
1A50  66 28 08 00 00 05 66 40  : 41
1A58  08 00 00 06 66 32 4A 86  : 76
1A60  66 24 22 2E 00 0C 67 40  : 8D
1A68  70 20 B2 7C 00 02 64 18  : 3C
1A70  4A AE 00 08 67 32 70 30  : 39
1A78  60 0E 4A 86 66 08 70 2B  : 47
-----------------------------------
SUM:  A6 79 DD 98 C3 C4 F4 BC  D703
```

▶Final Ver.3の紹介はたいへんタイムリーでした。WINDEXを持っていたのですが，Finalも思わず買ってしまったところだったのです。現在はFinal+E1で，日本語ワープロみたいにして使っています。
林　茂雄（24）三重県

```
1A80  60 06 70 20 60 02 70 2D  : F5
1A88  61 00 FE 16 10 80 4E 75  : C8
1A90  61 00 FE 24 70 20 60 06  : 79
1A98  61 00 FE 1C 70 2B 4A 86  : E6
1AA0  67 02 70 2D 10 C0 42 10  : 28
1AA8  4E 75 4A 47 6A 1E 44 47  : 67
1AB0  20 6E FF EA BE 6E FF EE  : 90
1AB8  65 06 3E 2E FF EE 67 0C  : 37
1AC0  61 00 FD DE 10 BC 00 30  : 38
1AC8  53 47 66 F4 4E 75 4A AE  : AF
1AD0  00 10 6B 10 20 6E FF EA  : 02
1AD8  41 F0 70 00 61 00 FD C2  : C1
1AE0  10 BC 00 2E 4E 75 4A 47  : 4E
1AE8  66 32 20 2E 00 0C 67 2C  : 85
1AF0  55 80 64 1A 20 2E 00 14  : B5
1AF8  08 00 00 04 66 10 08 00  : 8A
------------------------------------
SUM:  85 A6 23 5E 3A 65 53 90  5460

1B00  00 05 66 0A 08 00 00 06  : 83
1B08  66 04 4A 86 66 0E 20 6E  : 3C
1B10  FF EA 61 00 FD 8C 10 BC  : 9F
1B18  00 30 52 47 4E 75 08 2E  : C2
1B20  00 02 00 17 67 1C 20 6E  : 2A
1B28  FF EA 41 F0 70 00 57 88  : 69
1B30  B1 EE FF EA 63 0C 61 00  : 58
1B38  FD 68 10 BC 00 2C 52 47  : F6
1B40  60 EC 4E 75 20 6E FF EA  : 86
1B48  20 2E 00 14 08 00 00 04  : 6E
1B50  66 1C 08 00 00 05 66 1E  : 13
1B58  08 00 00 06 66 24 4A 86  : 68
1B60  67 0A 70 2D 61 00 FD 3A  : A6
1B68  10 80 52 47 4E 75 4A 86  : BC
1B70  66 F0 70 2B 60 EE 61 00  : A0
1B78  FD 3E 4A 86 66 10 70 2B  : 1C
------------------------------------
SUM:  DA 53 85 38 F6 6D 29 18  949A

1B80  60 0E 61 00 FD 32 4A 86  : CE
1B88  66 04 70 20 60 02 70 2D  : F9
1B90  10 C0 42 10 4E 75 08 2E  : 1B
1B98  00 01 00 17 67 0E 20 6E  : 1B
1BA0  FF EA 61 00 FC FC 10 BC  : 0E
1BA8  00 5C 52 47 4E 75 48 C7  : C7
1BB0  9E AE 00 04 64 1C 20 6E  : 5E
1BB8  FF EA 44 87 70 20 08 2E  : 7A
1BC0  00 00 00 17 67 02 70 2A  : 1A
1BC8  61 00 FC D6 10 80 53 87  : 9D
1BD0  66 F6 4E 75 2D 48 FF EA  : 7D
1BD8  42 6E FF EE 1D 42 FF EF  : EA
1BE0  61 6A 43 EE FF F0 12 FC  : F9
1BE8  00 30 2F 09 61 00 00 E8  : B1
1BF0  41 EE FF FF 61 00 01 12  : A1
1BF8  22 5F B1 C9 64 04 22 48  : CD
------------------------------------
SUM:  3F FC 75 28 16 64 58 36  BF3E

1C00  52 47 4E 75 4A 40 6B 16  : 67
1C08  B0 7C 00 0E 64 10 41 F1  : E0
1C10  00 00 61 00 00 F4 B1 C9  : CF
1C18  64 04 22 48 52 47 4E 75  : 2E
1C20  20 6E FF EA 4A 40 67 20  : 88
1C28  6B 1E 42 41 B2 7C 00 FF  : 39
1C30  67 16 B2 40 67 12 B2 7C  : 16
1C38  00 0E 64 06 10 D9 52 41  : F4
1C40  60 EA 10 FC 00 30 60 F6  : DC
1C48  42 10 4E 75 7C 00 24 00  : B5
1C50  02 80 7F FF FF FF 66 08  : 6C
1C58  4A 81 66 04 42 47 4E 75  : 81
1C60  D4 82 DD 86 7E 0F 24 3C  : A6
1C68  43 0C 6B F5 26 3C 26 34  : 6B
1C70  00 00 61 00 F0 46 64 24  : 1F
1C78  24 3C 3F F0 00 00 76 00  : 05
------------------------------------
SUM:  81 3C 53 1B C4 39 72 28  2FC0

1C80  61 00 F0 38 64 2C 24 3C  : 79
1C88  43 0C 6B F5 26 3C 26 34  : 6B
1C90  00 00 61 00 F2 DC 04 47  : 7A
1C98  00 0F 60 CA 24 3C 43 0C  : E8
1CA0  6B F5 26 3C 26 34 00 00  : 1C
1CA8  61 00 F4 0C 06 47 00 0F  : BD
1CB0  60 B4 24 3C 42 D6 BC C4  : 0C
1CB8  26 3C 1E 90 00 00 61 00  : 71
1CC0  EF FA 64 10 24 3C 40 24  : 21
1CC8  00 00 76 00 61 00 F2 A2  : 6B
1CD0  53 47 60 DE 4E 75 38 3C  : 0F
1CD8  00 0E 45 FA 00 4A 12 BC  : 65
1CE0  00 30 41 D2 24 10 26 28  : C5
1CE8  00 04 61 00 EF CE 65 0E  : 95
1CF0  52 11 24 10 26 28 00 04  : E9
1CF8  61 00 F1 5E 60 E6 52 89  : D1
------------------------------------
SUM:  EB 94 AE 33 7A B8 07 17  94B6

1D00  50 8A 51 CC FF DA 4E 75  : 93
1D08  0C 10 00 35 65 0E 52 20  : 36
1D10  0C 10 00 39 63 06 10 BC  : 8A
1D18  00 30 60 F2 4E 75 43 0C  : 94
1D20  6B F5 26 34 00 00 42 D6  : D2
1D28  BC C4 1E 90 00 00 42 A2  : 12
1D30  30 9C E5 40 00 00 42 6D  : A0
1D38  1A 94 A2 00 00 00 42 37  : C9
1D40  48 76 E8 00 00 00 42 02  : EA
1D48  A0 5F 20 00 00 00 41 CD  : 2D
1D50  CD 65 00 00 00 00 41 97  : 0A
1D58  D7 84 00 00 00 00 41 63  : FF
1D60  12 D0 00 00 00 00 41 2E  : 51
1D68  84 80 00 00 00 00 40 F8  : 3C
1D70  6A 00 00 00 00 00 40 C3  : 6D
1D78  88 00 00 00 00 00 40 8F  : 57
------------------------------------
SUM:  ED D1 84 30 15 63 01 BA  916C
```

```
1D80  40 00 00 00 00 00 40 59  : D9
1D88  00 00 00 00 00 00 40 24  : 64
1D90  00 00 00 00 00 00 3F F0  : 2F
1D98  00 00 00 00 00 00 2F 01  : 30
1DA0  48 7A E8 1C 20 3C 40 09  : 6B
1DA8  21 FB 22 3C 54 44 2D 18  : 57
1DB0  4E 75 2F 01 61 00 0A 32  : 90
1DB8  48 7A E8 04 48 E7 30 00  : 0D
1DC0  24 3C 40 09 21 FB 26 3C  : 27
1DC8  54 44 2D 18 61 00 F1 A2  : D1
1DD0  4C DF 00 0C 4E 75 48 E7  : 29
1DD8  30 00 48 7A FF F4 48 E7  : 14
1DE0  C0 00 20 02 22 03 61 00  : 68
1DE8  EE BA 67 24 61 00 EB DC  : 5B
1DF0  61 00 EE B0 67 38 4C DF  : C9
1DF8  00 03 61 00 EE A6 67 1C  : 7B
------------------------------------
SUM:  42 80 AC DA C4 AC 3B 44  99DE

1E00  61 00 02 2A 65 14 61 00  : 67
1E08  F1 68 65 0E 60 00 03 18  : 47
1E10  50 8F 20 3C 3F F0 00 00  : 6A
1E18  72 00 4E 75 4A 82 6A FA  : 65
1E20  20 3C 7F FF FF FF 72 FF  : 49
1E28  44 FC 00 05 4E 75 4C DF  : 33
1E30  00 03 48 E7 31 00 4E 56  : 07
1E38  00 00 48 E7 C0 00 2F 02  : 20
1E40  08 82 00 1F 20 02 22 03  : F0
1E48  61 00 EC 60 65 22 2E 00  : 62
1E50  61 48 65 1C 4A 9F 6A 10  : 8D
1E58  24 00 26 01 20 3C 3F F0  : D6
1E60  00 00 72 00 61 00 F2 50  : 15
1E68  4E 5E 4C DF 00 8C 4E 75  : 26
1E70  40 C0 4A AE FF F4 6A 04  : 59
1E78  08 40 00 01 08 00 00 01  : 52
------------------------------------
SUM:  FC 5A 63 E5 E3 79 AC 15  C6D9

1E80  66 0A 70 00 72 00 44 FC  : 92
1E88  00 01 60 DC 20 3C 7F FF  : 17
1E90  FF FF 72 FF 44 FC 00 03  : B2
1E98  60 CE 4A 87 67 28 08 07  : 9D
1EA0  00 00 67 14 53 87 67 24  : E0
1EA8  61 F0 65 C4 24 2E FF F8  : C3
1EB0  26 2E FF FC 60 00 F0 BA  : 59
1EB8  E2 8F 61 DE 65 B2 24 00  : EB
1EC0  26 01 60 00 F0 AC 42 80  : E5
1EC8  42 81 4E 75 20 2E FF F8  : CB
1ED0  22 2E FF FC 4E 75 2F 01  : 3E
1ED8  61 00 09 0E 48 7A E6 E0  : 00
1EE0  61 00 ED C0 67 4E 6B 52  : 80
1EE8  48 E7 30 00 24 00 42 42  : 07
1EF0  48 42 76 00 02 80 00 0F  : 91
1EF8  FF FF 02 42 7F F0 04 42  : F7
------------------------------------
SUM:  09 5D 03 95 2B 4E 4C 19  0BB1

1F00  3F F0 08 02 00 04 67 12  : B6
1F08  06 42 00 10 E2 42 06 42  : C4
1F10  3F F0 00 80 3F E0 00 00  : CE
1F18  60 0C E2 42 06 42 3F F0  : 07
1F20  00 80 3F F0 00 00 48 42  : 39
1F28  61 1A 61 00 F0 44 4C DF  : 3B
1F30  00 0C 4E 75 02 3C 00 FE  : 0B
1F38  4E 75 70 00 72 00 44 FC  : E5
1F40  00 01 4E 75 48 E7 3C 00  : 2F
1F48  28 00 2A 01 61 00 ED CA  : 6B
1F50  61 00 F1 12 48 E7 C0 00  : 53
1F58  24 00 26 01 20 04 22 05  : 96
1F60  61 00 F1 54 24 17 26 2F  : 36
1F68  00 04 61 00 ED F0 61 00  : A3
1F70  F0 F4 4C DF 00 0C 61 00  : 7C
1F78  ED 42 65 D8 4C DF 00 3C  : D3
------------------------------------
SUM:  7E 84 DA CD F9 AC 77 99  23A5

1F80  4E 75 48 E7 F0 00 4C D6  : 04
1F88  00 0F 48 7A E6 28 48 E7  : 0E
1F90  F0 00 61 00 F1 22 65 38  : 01
1F98  34 00 02 82 7F F0 00 00  : 27
1FA0  B4 BC 43 30 00 00 64 24  : 6B
1FA8  61 00 E9 44 24 2F 00 08  : E9
1FB0  26 2F 00 0C 61 00 EF BA  : 6B
1FB8  65 16 24 00 26 01 4C DF  : F1
1FC0  00 03 61 00 EE 94 4C DF  : 11
1FC8  00 0C 4E 75 44 FC 00 03  : 12
1FD0  4C DF 00 0F 4E 75 4A 80  : C7
1FD8  6A 1A B0 BC 80 00 00 00  : 70
1FE0  66 04 4A 81 67 0A 20 3C  : 02
1FE8  BF F0 00 00 72 00 4E 75  : E4
1FF0  70 00 4E 75 66 04 4A 81  : 68
1FF8  67 08 72 00 20 3C 3F F0  : 6C
------------------------------------
SUM:  C4 89 AC 99 50 B9 25 3E  DAF9

2000  00 00 4E 75 4A 80 67 E8  : DC
2008  6A 10 B0 BC 80 00 00 00  : 66
2010  67 DE 20 3C BF 80 00 00  : E0
2018  4E 75 20 3C 3F 80 00 00  : DE
2020  4E 75 2F 01 61 00 07 C2  : 1D
2028  48 7A E5 94 61 00 EC 74  : FC
2030  67 00 00 C0 6B 00 00 C6  : 58
2038  48 E7 3F E0 4E 56 00 00  : F2
2040  28 00 2A 01 48 40 E8 48  : 0B
2048  02 80 00 00 07 FF 04 80  : 0C
2050  00 00 03 FF 61 00 E9 EC  : 38
2058  24 3C 3F E6 2E 42 26 3C  : 57
2060  FE FA 39 EF 61 00 EF 0A  : 7A
2068  65 00 03 88 C1 44 C3 45  : FD
2070  48 40 C0 7C BF FF 80 7C  : 7E
2078  3F F0 48 40 61 10 65 00  : 8D
------------------------------------
SUM:  9C 1F 41 F7 63 AA EC 9F  A903
```

```
2080  03 72 24 04 26 05 61 00  : 29
2088  EC D4 60 00 03 66 24 3C  : E9
2090  3F F2 BE E0 76 00 61 00  : A6
2098  EC 22 65 00 03 D4 24 3C  : AA
20A0  3F FA 82 C0 76 00 61 00  : 52
20A8  EC 12 65 1C 04 80 00 10  : 13
20B0  00 00 61 00 03 BC 65 38  : BD
20B8  24 3C 3F E6 2E 42 26 3C  : 57
20C0  FE FA 39 EF 60 00 EC 96  : 02
20C8  24 3C 3F E6 A0 9E 26 3C  : 25
20D0  66 7F 3B CC 61 00 EE 9A  : D5
20D8  65 16 61 00 03 94 65 10  : E8
20E0  24 3C 3F D6 2E 42 26 3C  : 47
20E8  FE FA 39 EF 60 00 EC 6E  : DA
20F0  4E 75 70 00 72 00 44 FC  : E5
20F8  00 05 4E 75 70 00 72 00  : AA
------------------------------------
SUM:  C6 1D 78 81 21 31 23 1E  6B10

2100  44 FC 00 01 4E 75 4A 84  : D2
2108  6B 00 02 E4 20 3C 7F FF  : 2B
2110  FF FF 72 FF 44 FC 00 03  : B2
2118  60 00 02 D8 2F 01 61 00  : CB
2120  06 C8 48 7A E4 9A 48 E7  : 3D
2128  3F E0 4E 56 00 00 28 00  : EB
2130  24 3C 3F E6 2E 42 26 3C  : 57
2138  FE FA 39 EF 61 00 EF 78  : E8
2140  65 C4 28 00 2A 01 61 00  : DD
2148  E7 A6 24 3C 40 90 00 00  : BD
2150  76 00 61 00 EB 66 64 B4  : 40
2158  24 3C C0 90 00 00 76 00  : 26
2160  61 00 EB 58 63 00 02 88  : 91
2168  61 00 E9 40 2C 00 20 04  : DA
2170  22 05 61 00 E8 56 24 3C  : 26
2178  3F E6 2E 42 26 3C FE FA  : EF
------------------------------------
SUM:  7E 6A 54 07 46 13 2E 97  26CE

2180  39 EF 61 00 ED EC 65 24  : EB
2188  24 00 48 42 02 42 7F F0  : 61
2190  04 42 00 10 65 16 04 80  : 55
2198  00 10 00 00 61 00 02 C8  : 3B
21A0  65 0A 24 00 26 01 61 00  : 1B
21A8  ED C8 64 08 20 3C 3F F0  : AC
21B0  00 00 72 00 24 00 48 42  : 20
21B8  E8 4A 02 82 00 00 07 FF  : BC
21C0  D4 46 6B 00 02 2A B4 7C  : E1
21C8  07 FF 64 00 FF 40 C0 BC  : 25
21D0  80 0F FF FF E9 4A 48 42  : 4A
21D8  42 42 80 82 60 00 02 14  : FC
21E0  2F 01 61 00 06 04 48 7A  : 5D
21E8  E3 D6 48 E7 3F E0 4E 56  : AB
21F0  00 00 2F 00 08 80 00 1F  : D6
21F8  24 3C 3F F0 00 00 76 00  : 05
------------------------------------
SUM:  6E 06 0A 34 B6 99 A3 0A  30E9

2200  61 00 EA B8 65 24 C1 42  : 8F
2208  C3 43 61 00 EE AA 61 24  : 84
2210  65 00 01 E0 24 00 26 01  : 91
2218  20 3C 3F F9 21 FB 22 3C  : 0E
2220  54 44 2D 18 61 00 EC 32  : 5C
2228  60 02 61 08 64 00 01 94  : C4
2230  60 00 01 C0 24 3C 3F C9  : 89
2238  75 A0 76 00 61 00 EA 7C  : 52
2240  65 00 01 E0 24 3C 3F E5  : CA
2248  61 20 76 00 61 00 EA 6C  : AE
2250  64 54 28 00 2A 01 24 3C  : 6B
2258  3F DA 82 79 26 3C 99 FC  : 0B
2260  EF 32 61 00 ED 0C 65 00  : E0
2268  00 80 61 00 EA AC 65 78  : 54
2270  C1 44 C3 45 24 3C 3F DA  : 86
2278  82 79 26 3C 99 FC EF 32  : 13
------------------------------------
SUM:  CD 22 5C 4B 4B 6E 5E BB  B65E

2280  61 00 EB D6 65 62 24 04  : 11
2288  26 05 61 00 EE 2A 65 58  : 61
2290  61 00 01 90 65 52 24 3C  : 09
2298  3F D9 21 FB 26 3C 54 44  : 2E
22A0  2D 18 60 00 EA B8 28 00  : 6F
22A8  2A 01 61 00 EA 6C 65 38  : 7F
22B0  24 04 26 05 28 00 2A 01  : A6
22B8  20 3C 3F F0 00 00 72 00  : FD
22C0  61 00 EB 96 65 22 24 04  : 91
22C8  26 05 61 00 ED EA 65 18  : E0
22D0  61 00 01 50 24 00 26 01  : FD
22D8  20 3C 3F E9 21 FB 22 3C  : FE
22E0  54 44 2D 18 60 00 EB 72  : 9A
22E8  4E 75 2F 01 61 00 04 FA  : 52
22F0  48 7A E2 CC 48 E7 3F E0  : BE
22F8  4E 56 00 00 24 00 26 01  : EF
------------------------------------
SUM:  02 01 5E 0A 9E 2C 4F BB  45EA

2300  61 20 65 00 00 EE C1 42  : D7
2308  C3 43 61 46 65 00 00 E4  : F6
2310  61 00 ED A4 60 00 00 DC  : 2E
2318  2F 01 61 00 04 CC 48 7A  : 23
2320  E2 9E 48 E7 3F E0 4E 56  : 72
2328  00 00 61 00 00 A0 24 3C  : 61
2330  3F F9 21 FB 26 3C 54 44  : 4E
2338  2D 18 61 00 EB 1C 65 00  : 12
2340  00 B2 61 00 E9 AE 60 14  : 1E
2348  2F 01 61 00 04 9C 48 7A  : F3
2350  E2 6E 48 E7 3F E0 4E 56  : 42
2358  00 00 61 70 2F 00 08 80  : 88
2360  00 1F 24 3C 40 19 21 FB  : F4
2368  26 3C 54 44 2D 18 61 00  : A0
2370  E9 4A 65 06 61 00 FC 18  : 13
2378  65 78 24 3C 40 09 21 FB  : A2
------------------------------------
SUM:  87 51 AB E5 82 F6 D1 C4  7030
```

▶先日，光栄より「水滸伝」と「信長の野望・戦国群雄伝」のPC-88版が出ました。3月までにX1に移植されたら，受験生がかわいそうだとは思いながらも期待しています。

田中　英伸（17）栃木県

```
2380  26 3C 54 44 2D 18 61 00  : A0
2388  E9 32 65 0A 61 00 EA CA  : 9F
2390  65 60 08 57 00 07 24 3C  : 8B
2398  3F F9 21 FB 26 3C 54 44  : 4E
23A0  2D 18 61 00 E9 16 65 16  : 20
23A8  24 3C 40 09 21 FB 26 3C  : 27
23B0  54 44 2D 18 61 00 EA A2  : CA
23B8  65 38 08 80 00 1F 61 3A  : DF
23C0  65 30 4A 9F 6A 2C 08 C0  : DC
23C8  00 1F 60 26 4A 80 6B 10  : EA
23D0  24 3C 41 97 D7 84 76 00  : 09
23D8  61 00 E8 E0 64 10 4E 75  : 60
23E0  24 3C C1 97 D7 84 76 00  : 89
23E8  61 00 E8 D0 62 0A 70 00  : F5
23F0  72 00 4E 5E 4C DF 07 FC  : 4C
23F8  4E 75 24 3C 3F E9 21 FB  : 67
-----------------------------------------
SUM:  EC D3 A6 7E D2 21 DE B4  0AD0

2400  26 3C 54 44 2D 18 61 00  : A0
2408  E8 B2 65 20 24 3C 3F F9  : B7
2410  21 FB 26 3C 54 44 2D 18  : 5B
2418  61 00 EA 3E 08 80 00 1F  : 30
2420  60 30 41 FA 00 FC 43 FA  : 04
2428  01 48 60 08 41 FA 00 72  : 5E
2430  43 FA 00 AE 28 00 2A 01  : 3E
2438  61 08 64 04 20 04 22 05  : 1C
2440  4E 75 48 E7 C0 00 61 12  : 25
2448  65 52 4C DF 00 0C 60 00  : 4E
2450  EB 20 41 FA 00 8C 43 FA  : 0F
2458  00 C8 24 00 26 01 61 00  : 74
2460  EB 10 64 18 4E 75 41 FA  : 75
2468  01 08 43 FA 01 64 60 0C  : 17
2470  61 00 E9 96 41 FA 01 5A  : 76
2478  43 FA 01 EE 48 E7 C0 00  : 1B
-----------------------------------------
SUM:  C3 24 58 E8 F4 65 23 0E  0447

2480  4C D8 00 03 4C D7 00 0C  : 56
2488  61 00 EA E6 65 0E 4C D8  : C8
2490  00 0C 61 00 E8 C8 65 04  : 86
2498  B1 C9 66 E8 50 8F 4E 75  : 6A
24A0  BD 6A E7 F3 E7 33 B8 1F  : F2
24A8  3D E6 12 46 13 A8 6D 09  : AC
24B0  BE 5A E6 45 67 F5 44 E4  : C7
24B8  3E C7 1D E3 A5 56 C7 34  : FB
24C0  BF 2A 01 A0 1A 01 A0 1A  : 5F
24C8  3F 81 11 11 11 11 11 11  : 26
24D0  BF C5 55 55 55 55 55 55  : 82
24D8  3F F0 00 00 00 00 00 00  : 2F
24E0  BD A9 39 74 A8 C0 7C 9D  : 94
24E8  3E 21 EE D8 EF F8 D8 98  : 7C
24F0  BE 92 7E 4F B7 78 9F 5C  : 47
24F8  3E FA 01 A0 1A 01 A0 1A  : AE
-----------------------------------------
SUM:  47 D4 BA 73 D7 FA C8 C8  C006

2500  BF 56 C1 6C 16 C1 6C 17  : 9C
2508  3F A5 55 55 55 55 55 55  : E2
2510  BF E0 00 00 00 00 00 00  : 9F
2518  3F F0 00 00 00 00 00 00  : 2F
2520  BF AA F2 86 BC A1 AF 28  : 15
2528  3F AE 1E 1E 1E 1E 1E 1E  : A1
2530  BF B1 11 11 11 11 11 11  : D6
2538  3F B3 B1 3B 13 B1 3B 14  : F1
2540  BF B7 45 D1 74 5D 17 46  : BA
2548  3F BC 71 C7 1C 71 C7 1C  : A3
2550  BF C2 49 24 92 49 24 92  : 7F
2558  3F C9 99 99 99 99 99 9A  : 9F
2560  BF D5 55 55 55 55 55 55  : 92
2568  3F F0 00 00 00 00 00 00  : 2F
2570  3E 5A E6 45 67 F5 44 E4  : 47
2578  3E 92 7E 4F B7 78 9F 5C  : C7
-----------------------------------------
SUM:  6E 36 39 EF 97 09 AD FA  6DC6

2580  3E C7 1D E3 A5 56 C7 34  : FB
2588  3E FA 01 A0 1A 01 A0 1A  : AE
2590  3F 2A 01 A0 1A 01 A0 1A  : DF
2598  3F 56 C1 6C 16 C1 6C 17  : 1C
25A0  3F 81 11 11 11 11 11 11  : 26
25A8  3F A5 55 55 55 55 55 55  : E2
25B0  3F C5 55 55 55 55 55 55  : 02
25B8  3F E0 00 00 00 00 00 00  : 1F
25C0  3F F0 00 00 00 00 00 00  : 2F
25C8  3F F0 00 00 00 00 00 00  : 2F
25D0  BF AC 71 C7 1C 71 C7 1C  : 13
25D8  3F AE 1E 1E 1E 1E 1E 1E  : A1
25E0  BF B0 00 00 00 00 00 00  : 6F
25E8  3F B1 11 11 11 11 11 11  : 56
25F0  BF B2 49 24 92 49 24 92  : 6F
25F8  3F B3 B1 3B 13 B1 3B 14  : F1
-----------------------------------------
SUM:  6E 0C 35 9F 9A 6E 83 2B  277D

2600  BF B5 55 55 55 55 55 55  : 72
2608  3F B7 45 D1 74 5D 17 46  : 3A
2610  BF B9 99 99 99 99 99 9A  : 0F
2618  3F BC 71 C7 1C 71 C7 1C  : A3
2620  BF C0 00 00 00 00 00 00  : 7F
2628  3F C2 49 24 92 49 24 92  : FF
2630  BF C5 55 55 55 55 55 55  : 82
2638  3F C9 99 99 99 99 99 9A  : 9F
2640  BF D0 00 00 00 00 00 00  : 8F
2648  3F D5 55 55 55 55 55 55  : 12
2650  BF E0 00 00 00 00 00 00  : 9F
2658  3F F0 00 00 00 00 00 00  : 2F
2660  00 00 00 00 00 00 00 00  : 00
2668  06 80 00 00 80 00 B0 BC  : 72
2670  00 01 00 00 65 04 70 FF  : D9
2678  4E 75 2F 01 02 80 00 00  : 75
-----------------------------------------
SUM:  48 5C 5F EE 3A CC 53 E2  4640

2680  FF FF E5 88 72 01 08 80  : 66
2688  00 02 67 02 72 03 80 81  : E1
2690  23 C0 00 00 26 96 4C DF  : CA
2698  00 02 42 80 4E 75 61 08  : F0
26A0  02 80 00 00 7F FF 4E 75  : C3
26A8  48 E7 40 80 41 FA 00 28  : 52
26B0  20 10 2F 00 22 3C 00 00  : BD
26B8  03 83 61 00 E4 CA 02 80  : 17
26C0  00 03 FF FF 20 80 20 1F  : E0
26C8  72 00 E2 88 E2 88 D1 41  : 58
26D0  4C DF 01 02 4E 75 00 00  : F1
26D8  00 01 48 E7 C0 00 4C D6  : 12
26E0  00 03 48 7A DE C6 48 E7  : 98
26E8  71 00 2E 00 67 0A 4A 81  : DB
26F0  67 0C 61 00 E5 8C 20 02  : 67
26F8  4C DF 00 8E 4E 75 42 80  : 3E
-----------------------------------------
SUM:  71 8E 5F 02 A6 5C B6 25  CF8F

2700  4C DF 00 8E 00 3C 00 01  : F6
2708  4E 75 2F 00 20 16 48 7A  : EA
2710  DE 90 4A 80 67 48 48 E7  : 16
2718  C0 00 02 97 80 00 00 00  : D9
2720  32 3C 4B 00 4A 80 6A 02  : EF
2728  44 80 B0 BC 00 80 00 00  : B0
2730  64 14 08 00 00 17 66 16  : 13
2738  04 41 00 80 D0 80 60 F2  : 67
2740  E2 88 06 41 00 80 B0 BC  : 9D
2748  01 00 00 00 64 F2 48 41  : E0
2750  42 41 08 80 00 17 80 81  : 23
2758  80 9F 4C DF 00 02 4E 75  : 0F
2760  2F 00 20 16 48 7A DE 3A  : 3F
2768  48 E7 60 00 22 00 48 41  : 3A
2770  EE 49 92 3C 00 7F 65 3E  : 27
2778  B0 3C 00 1F 64 40 02 41  : F2
-----------------------------------------
SUM:  D0 C9 EA F2 53 F5 13 59  B095

2780  00 FF 24 00 08 C0 00 17  : 02
2788  02 80 00 00 FF FF B2 7C  : AD
2790  00 17 64 1A E2 88 52 41  : 92
2798  B2 7C 00 17 65 F6 4A 82  : 6C
27A0  6A 02 44 80 4C DF 00 06  : 61
27A8  4E 75 D0 80 53 41 B2 7C  : D5
27B0  00 17 66 F6 60 E8 4C DF  : E6
27B8  00 06 70 00 4E 75 4C DF  : 64
27C0  00 06 20 3C 7F FF FF FF  : DE
27C8  44 FC 00 03 4E 75 48 E7  : 35
27D0  C0 00 4C D6 00 03 48 7A  : A7
27D8  DD D2 60 0C C1 41 28 01  : 46
27E0  61 06 24 00 26 01 20 04  : D6
27E8  4A 80 67 24 B0 BC 80 00  : 41
27F0  00 00 67 1C 72 00 E2 80  : 57
27F8  E2 91 E2 80 E2 91 E2 80  : AA
-----------------------------------------
SUM:  DA 91 12 07 53 C0 B3 FB  3EC9

2800  E2 91 48 40 02 40 8F FF  : CB
2808  06 40 38 00 48 40 4E 75  : C9
2810  42 81 4E 75 48 E7 C0 00  : 75
2818  4C D6 00 03 48 7A DD 8C  : 50
2820  40 E7 48 E7 20 00 65 3E  : 19
2828  24 00 48 42 02 42 7F F0  : 61
2830  67 2C 04 42 38 00 B4 7C  : 41
2838  48 00 64 38 E9 4A D2 81  : 6A
2840  D1 80 E2 52 48 42 42 42  : 93
2848  D2 81 D1 80 D2 81 D1 80  : 48
2850  02 80 00 7F FF FF 80 82  : 01
2858  24 1F 44 DF 4E 75 24 1F  : 6C
2860  44 DF 42 80 4E 75 67 2C  : 3B
2868  6B 3A 69 0C 61 00 E4 34  : 93
2870  67 16 60 B4 4A 42 6B 10  : 98
2878  24 1F 44 DF 20 3C 7F FF  : 40
-----------------------------------------
SUM:  8C 29 0C AA 9D 97 D0 FD  BF90

2880  FF FF 44 FC 00 03 4E 75  : 04
2888  24 1F 70 00 54 4F 44 FC  : 96
2890  00 01 4E 75 24 1F 54 4F  : AA
2898  20 3C 7F FF FF FF 44 FC  : 18
28A0  00 05 4E 75 24 1F 54 4F  : AE
28A8  44 FC 00 09 4E 75 00 06  : 12
28B0  00 04 00 12 00 36 00 4A  : 96
28B8  00 C4 00 04 00 04 00 04  : D0
28C0  00 04 00 04 00 04 00 04  : 10
28C8  00 04 00 04 00 04 00 04  : 10
28D0  00 04 00 04 00 04 00 04  : 10
28D8  00 04 00 04 00 04 00 04  : 10
28E0  00 04 00 04 00 04 00 04  : 10
28E8  00 04 00 04 00 04 00 04  : 10
28F0  00 04 00 04 00 04 00 04  : 10
28F8  00 04 00 04 00 04 00 04  : 10
-----------------------------------------
SUM:  87 44 CF 24 E9 5E 7E 7F  F4BD

2900  00 04 00 04 00 04 00 04  : 10
2908  00 04 00 04 00 04 00 04  : 10
2910  00 04 00 04 00 04 00 04  : 10
2918  00 04 00 04 00 04 00 04  : 10
2920  00 04 00 04 00 04 00 04  : 10
2928  00 04 00 04 00 04 00 04  : 10
2930  00 04 00 04 00 04 00 04  : 10
2938  00 04 00 04 00 04 00 04  : 10
2940  00 04 00 04 00 04 00 04  : 10
2948  00 04 00 04 00 04 00 04  : 10
2950  00 04 00 04 00 04 00 04  : 10
2958  00 04 00 04 00 04 00 04  : 10
2960  00 04 00 04 00 04 00 04  : 10
2968  00 04 00 04 00 04 00 04  : 10
2970  00 04 00 04 00 04 00 04  : 10
2978  00 04 00 04 00 04 00 04  : 10
-----------------------------------------
SUM:  00 40 00 40 00 40 00 40  A83D

2980  00 04 00 04 00 04 00 04  : 10
2988  00 04 00 04 00 04 00 04  : 10
2990  00 04 00 04 00 04 00 04  : 10
2998  00 04 00 04 00 04 00 04  : 10
29A0  00 04 00 04 00 04 00 04  : 10
29A8  00 04 00 04 00 04 00 04  : 10
29B0  00 04 00 04 00 04 00 04  : 10
29B8  00 04 00 04 00 04 00 04  : 10
29C0  00 04 00 04 00 04 00 04  : 10
29C8  00 04 00 04 00 04 00 04  : 10
29D0  00 04 00 04 00 04 00 04  : 10
29D8  00 04 00 04 00 04 00 04  : 10
29E0  00 04 00 04 00 04 00 04  : 10
29E8  00 04 00 04 00 04 00 04  : 10
29F0  00 04 00 04 00 04 00 04  : 10
29F8  00 04 00 04 00 04 00 04  : 10
-----------------------------------------
SUM:  00 40 00 40 00 40 00 40  A83D

2A00  00 04 00 04 00 04 00 04  : 10
2A08  00 04 00 04 00 04 00 04  : 10
2A10  00 04 00 04 00 04 00 04  : 10
2A18  00 04 00 04 00 04 00 04  : 10
2A20  00 04 00 04 00 04 00 04  : 10
2A28  00 04 00 04 00 04 00 04  : 10
2A30  00 04 00 04 00 04 00 04  : 10
2A38  00 04 00 04 00 04 00 04  : 10
2A40  00 04 00 04 00 04 00 04  : 10
2A48  00 04 00 04 00 04 00 04  : 10
2A50  00 04 00 04 00 04 00 04  : 10
2A58  00 04 00 04 00 04 00 04  : 10
2A60  00 04 00 04 00 04 00 04  : 10
2A68  00 04 00 04 00 04 00 04  : 10
2A70  00 04 00 04 00 04 00 04  : 10
2A78  00 04 00 04 00 04 00 04  : 10
-----------------------------------------
SUM:  00 40 00 40 00 40 00 40  A83D

2A80  00 04 00 04 00 04 00 04  : 10
2A88  00 04 00 04 00 04 00 04  : 10
2A90  00 04 00 04 00 04 00 04  : 10
2A98  00 04 00 04 00 04 00 04  : 10
2AA0  00 04 00 04 00 04 00 04  : 10
2AA8  00 04 00 04 00 04 00 04  : 10
2AB0  00 04 00 04 00 04 00 04  : 10
2AB8  20 F6 00 00 00 00 00 00  : 16
2AC0  00 00 00 00 00 00 00 00  : 00
2AC8  00 00 00 00 00 00 00 00  : 00
2AD0  00 00 00 00 00 00 00 00  : 00
2AD8  00 00 00 00 00 00 00 00  : 00
2AE0  00 00 00 00 00 00 00 00  : 00
2AE8  00 00 00 00 00 00 00 00  : 00
2AF0  00 00 00 00 00 00 00 00  : 00
2AF8  00 00 00 00 00 00 00 00  : 00
-----------------------------------------
SUM:  20 12 00 1C 00 1C 00 1C  3FB5
```

リスト3　ソースリスト(参考)

```
 1:         .NLIST
 2: * REWRITED BY M.YAMAGUCHI
 3:
 4: LEVEL   SET     4
 5: IEEE    SET     4
 6: **SHARP SET     0
 7:
 8: MCLRL   MACRO   P1
 9:         MOVEQ.L #0,P1
10:         ENDM
11:
12: IE_CONV MACRO   P1,P2
13: P_H     SET     ((P1.ASR.11).AND.$800FFFFF)+((P2.SHL.20).AND.$7FF00000)
14: P_L     SET     ((P1.SHL.21).AND.$FFE00000)+((P2.SHR.11).AND.$1FFFFF)
15:         ENDM
16:
17: IE_DL   MACRO   P1,P2
18:         IE_CONV P1,P2
19:         DC.L    P_H,P_L
20:         ENDM
21:
22: IE_REG  MACRO   P1,P2,P3,P4
23:         IE_CONV P1,P2
24:         IF P_H.EQ.0
25:          MOVEQ.L        #0,P3
26:         ELSE
27:          IF P_H.EQ.$FFFFFFFF
28:           MOVEQ.L       #$FF,P3
29:          ELSE
30:           MOVE.L        #P_H,P3
31:          ENDIF
32:         ENDIF
33:
34:         IF P_L.EQ.0
35:          MOVEQ.L        #0,P4
36:         ELSE
37:          IF P_L.EQ.$FFFFFFFF
38:           MOVEQ.L       #$FF,P4
39:          ELSE
40:           MOVE.L        #P_L,P4
41:          ENDIF
42:         ENDIF
```

▶祝！　自動車学校卒業。期間５カ月。所要時間50時間(23時間オーバー)。疲れた。

浅野　淳一 (19) 宮城県

```
 43:        ENDM
 44:
 45: FDCMP  MACRO  P1
 46:        CMP.L  D2,D0
 47:         BNE.B P1
 48:        CMP.L  D3,D1
 49:        ENDM
 50:
 51:        .TEXT
 52:
 53: FLOAT ST:
 54: * HEADER *
 55:        DC.L   -1
 56:        DC.W   $8000
 57:        DC.L   ENTRY
 58:        DC.L   INT ENTRY
 59:        DC.B   "FLOAT*/-"
 60:
 61: REQ_HEAD:
 62:        DC.L   0
 63: *ストラテジールーチン
 64: ENTRY:
 65:        MOVE.L A5,REQ_HEAD
 66:        RTS
 67: *割り込みルーチン
 68: INT_ENTRY:
 69:        MOVEM.L D0/A5,-(SP)
 70:        MOVE.L REQ_HEAD(PC),A5
 71:        MCLRL  D0              ;
 72:        MOVE.B 02(A5),D0       ;コマンドコード
 73:         BSR.B INT ESUB
 74:        MOVE.B D0,03(A5)       ;エラーコードLOW
 75:        LSR.W  #8,D0
 76:        MOVE.B D0,04(A5)       ;エラーコードHIGH
 77:        MOVEM.L (SP)+,D0/A5
 78:        RTS
 79:
 80: INT_ESUB:
 81:         BNE.B INT ESO
 82:        MOVEM.L D1/A1,-(SP)    ;コマンドコードが0の
時　初期化
 83:         BSR   SET VEC
 84:        MOVEM.L (SP)+,D1/A1
 85:         BEQ.B INT ESO
 86:        MOVE.L #COMN_BUF,$0E(A5)     ;デバイスドライバ
ーの終わりアドレス
 87:        CLR.L  D0
 88:        RTS
 89:
 90: INT_ESO:
 91:        MOVE.W #$5003,D0
 92:        RTS
 93:
 94: PAST_VEC:
 95:        .DC.L  0
 96:
 97: *1
 98: *処理ルーチンを置き換える　D0:FE番
号　A0:処理アドレス
 99: _FEVECS:
100:        CMP.L  #$0000FE00,D0
101:         BCS.B NO FEVEC
102:        CMP.L  #$0000FF00,D0
103:         BCC.B NO FEVEC        ;エリアチェック
104:        MOVE.L A1,-(SP)        ;MOVEM.L
105:        SUBI.W #$FE00,D0
106:        ASL.L  #2,D0
107:        LEA.L  FUNC TBL(PC),A1
108:        ADDA.L D0,A1
109:        MOVE.L (A1),D0         ;古いアドレスを返す
110:        MOVE.L A0,(A1)         ;新しいアドレスを設定
111:        MOVE.L (SP)+,A1
112:        RTS
113:
114: NO_FEVEC:
115:        MOVEQ.L #$FF,D0
116:        RTS
117:
118: SET_VEC:
119:        MOVEQ.L #$0B,D1        ;F 系列全て
120:        LEA.L  VEC ENTRY(PC),A1
121:        MOVEQ.L #$80,D0        ;B_INTVCS ベクタの設定
122:        TRAP   #15
123:        MOVE.L D0,PAST_VEC     ;設定前の処理アドレス
を待避
124:
125:        MOVE.L D0,A1
126:        SUBQ.L #4,A1
127:        MOVEQ.L #$84,D0        ;B_LPEEK 1ロングワードデ
ータ読み込み
128:        TRAP   #15
129:        CMP.L  ID_TBL(PC),D0   ;登録してあるのと同じ
かどうか
130:         BNE.B PR_COPY
131:
132:        MOVE.L PAST VEC(PC),A1
133:        MOVEQ.L #$0B,D1
134:        MOVEQ.L #$80,D0        ;B_INTVCS ベクタを元に戻す

135:        TRAP   #15
136:
137:        PEA.L  M STAY(PC)      ;すでに…
138:        .DC.W  $FF09           ;_PRINT
139:        ADDQ.L #4,SP
140:        CLR.L  D0
141:        RTS
142:
143: PR_COPY:
144:        PEA.L  M COPYRIGHT(PC) ;ライセンス表示
145:        .DC.W  $FF09           ;_PRINT
146:        ADDQ.L #4,SP
147:        MOVEQ.L #$FF,D0
148:        RTS
149: *コマンドラインから起動された時
150: START:
151:         BSR.B SET VEC         ;ベクター設定
152:         BEQ.B EXIT
153:        CLR.W  -(SP)
154:        MOVE.L #FLOAT_END-FLOAT ST,-(SP)      ;2F80H
155:        .DC.W  $FF31           ;_KEEPPR
156:
157: EXIT:  .DC.W  $FF00           ;_EXIT
158:
159: M_STAY:
160:        DC.B   $0D,$0A,"すでに浮動小数点演算"
161:        DC.B   "パッケージは登録されています",$0
D,$0A,0,0
162:        .EVEN
163: ID_TBL:
164:        DC.B   "FEfn"
165: *ベクター入り口
166: VEC_ENTRY:
167:        MOVEM.L D6/A5-A6,-(SP)
168:        LEA.L  $000E(SP),A6    ;F系列命令アドレス
169:        MOVE.L (A6),A5
170:        MOVE.W (A5)+,D6        ;ベクター番号
171:        SUB.W  #$FE00,D6
172:        CMPI.W #$100,D6        ;演算？
173:         BCC.B EX PAST
174:        MOVE.L A5,(A6)+        ;帰り先更新
175:
176:        BTST   #5,$000C(SP)    ;SYSTEM BITS
177:         BNE.B FLOAT 0
178:        MOVE   USP,A5          ;USER MODE
179: FLOAT_0:
180:        ASL.W  #2,D6           ;4BYTES
181:        MOVE.L FUNC_TBL-$(PC,D6.W),A5 ;処理アドレスをA
5に読み込む
182:        JSR    (A5)            ;テーブルジャンプ　演
算
183:        MOVE   SR,D6
184:        MOVE.B D6,$000D(SP)    ;FLAG CCR
185:        MOVEM.L (SP)+,D6/A5-A6
186:         IF LEVEL<4
187:         BTST  #7,(SP)
188:         BNE.B FLOAT_9
189:         ENDIF
190:        RTE
191:
192: FLOAT_9:
193:        ORI.W  #$8000,SR
194:        RTE
195:
196: EX_PAST:
197:        MOVEM.L (SP)+,D6/A5-A6
198:        MOVE.L PAST_VEC(PC),-(SP)     ;前のベクター実行
199: _NO:   RTS
200: FUNC_TBL:
201:        .DC.L  _LMUL
202:        .DC.L  _LDIV
203:        .DC.L  _LMOD
204:        .DC.L  _NO
205:        .DC.L  _UMUL
206:        .DC.L  _UDIV
207:        .DC.L  _UMOD
208:        .DC.L  _NO
209:        .DC.L  _IMUL
210:        .DC.L  _IDIV
211:        .DC.L  _NO
212:        .DC.L  _NO
213:        .DC.L  _RANDOMIZE
214:        .DC.L  _SRAND
215:        .DC.L  _RAND
216:        .DC.L  _NO
217:        .DC.L  _STOL
218:        .DC.L  _LTOS
219:        .DC.L  _STOH
220:        .DC.L  _HTOS
221:        .DC.L  _STOO
222:        .DC.L  _OTOS
223:        .DC.L  _STOB
224:        .DC.L  _BTOS
225:        .DC.L  _IUSING
226:        .DC.L  _NO
227:        .DC.L  _LTOD
228:        .DC.L  _DTOL
229:        .DC.L  _LTOF
230:        .DC.L  _FTOL
231:        .DC.L  _FTOD
232:        .DC.L  _DTOF
233:        .DC.L  _VAL
234:        .DC.L  _USING
235:        .DC.L  _STOD
236:        .DC.L  _DTOS
237:        .DC.L  _ECVT
238:        .DC.L  _FCVT
239:        .DC.L  _GCVT
240:        .DC.L  _NO
241:        .DC.L  _DTST
242:        .DC.L  _DCMP
243:        .DC.L  _DNEG
244:        .DC.L  _DADD
245:        .DC.L  _DSUB
246:        .DC.L  _DMUL
247:        .DC.L  _DDIV
248:        .DC.L  _DMOD
249:        .DC.L  _DABS
250:        .DC.L  _DCEIL
251:        .DC.L  _DFIX
252:        .DC.L  _DFLOOR
253:        .DC.L  _DFRAC
254:        .DC.L  _DSGN
255:        .DC.L  _SIN
256:        .DC.L  _COS
257:        .DC.L  _TAN
258:        .DC.L  _ATAN
259:        .DC.L  _LOG
260:        .DC.L  _EXP
261:        .DC.L  _SQR
262:        .DC.L  _PI
263:        .DC.L  _NPI
264:        .DC.L  _POWER
265:        .DC.L  _RND
266:        .DC.L  _NO
267:        .DC.L  _NO
268:        .DC.L  _NO
269:        .DC.L  _NO
270:        .DC.L  _NO
271:        .DC.L  _NO
272:        .DC.L  _NO
273:        .DC.L  _NO
274:        .DC.L  _DFREXP
275:        .DC.L  _DLDEXP
276:        .DC.L  _DADDONE
277:        .DC.L  _DSUBONE
278:        .DC.L  _DDIVTWO
279:        .DC.L  _NO2
280:        .DC.L  _NO2
281:        .DC.L  _FVAL
282:        .DC.L  _FUSING
283:        .DC.L  _STOF
284:        .DC.L  _FTOS
285:        .DC.L  _FECVT
286:        .DC.L  _FFCVT
287:        .DC.L  _FGCVT
288:        .DC.L  _NO
289:        .DC.L  _FTST
290:        .DC.L  _FCMP
291:        .DC.L  _FNEG
292:        .DC.L  _FADD
293:        .DC.L  _FSUB
294:        .DC.L  _FMUL
295:        .DC.L  _FDIV
296:        .DC.L  _FMOD
297:        .DC.L  _FABS
298:        .DC.L  _FCEIL
299:        .DC.L  _FFIX
300:        .DC.L  _FFLOOR
301:        .DC.L  _FFRAC
302:        .DC.L  _FSGN
303:        .DC.L  _FSIN
304:        .DC.L  _FCOS
305:        .DC.L  _FTAN
306:        .DC.L  _FATAN
307:        .DC.L  _FLOG
308:        .DC.L  _FEXP
309:        .DC.L  _FSQR
310:        .DC.L  _FPI
311:        .DC.L  _FNPI
312:        .DC.L  _FPOWER
313:        .DC.L  _FRND
314:        .DC.L  _NO
315:        .DC.L  _NO
316:        .DC.L  _NO
317:        .DC.L  _NO
318:        .DC.L  _NO
319:        .DC.L  _NO
320:        .DC.L  _NO
321:        .DC.L  _NO
322:        .DC.L  _FFREXP
323:        .DC.L  _FLDEXP
324:        .DC.L  _FADDONE
325:        .DC.L  _FSUBONE
326:        .DC.L  _FDIVTWO
327:        .DC.L  _NO2
328:        .DC.L  _NO2
329:        .DC.L  _NO
330:        .DC.L  _NO
331:        .DC.L  _NO
332:        .DC.L  _NO
333:        .DC.L  _NO
334:        .DC.L  _NO
335:        .DC.L  _NO
336:        .DC.L  _NO
337:        .DC.L  _NO
338:        .DC.L  _NO
339:        .DC.L  _NO
340:        .DC.L  _NO
341:        .DC.L  _NO
342:        .DC.L  _NO
343:        .DC.L  _NO
344:        .DC.L  _NO
345:        .DC.L  _NO
346:        .DC.L  _NO
347:        .DC.L  _NO
348:        .DC.L  _NO
349:        .DC.L  _NO
350:        .DC.L  _NO
351:        .DC.L  _NO
352:        .DC.L  _NO
353:        .DC.L  _NO
354:        .DC.L  _NO
355:        .DC.L  _NO
356:        .DC.L  _NO
357:        .DC.L  _NO
358:        .DC.L  _NO
359:        .DC.L  _NO
360:        .DC.L  _NO
361:        .DC.L  _NO
362:        .DC.L  _NO
363:        .DC.L  _NO
364:        .DC.L  _NO
365:        .DC.L  _NO
366:        .DC.L  _NO
367:        .DC.L  _NO
368:        .DC.L  _NO
369:        .DC.L  _NO
370:        .DC.L  _NO
371:        .DC.L  _NO
372:        .DC.L  _NO
373:        .DC.L  _NO
374:        .DC.L  _NO
375:        .DC.L  _NO
376:        .DC.L  _NO
377:        .DC.L  _NO
378:        .DC.L  _NO
379:        .DC.L  _NO
380:        .DC.L  _NO
381:        .DC.L  _NO
382:        .DC.L  _NO
383:        .DC.L  _NO
384:        .DC.L  _NO
385:        .DC.L  _NO
386:        .DC.L  _NO
387:        .DC.L  _NO
388:        .DC.L  _NO
389:        .DC.L  _NO
390:        .DC.L  _NO
391:        .DC.L  _NO
392:        .DC.L  _NO
393:        .DC.L  _NO
394:        .DC.L  _NO
395:        .DC.L  _NO
396:        .DC.L  _NO
397:        .DC.L  _NO
398:        .DC.L  _NO
399:        .DC.L  _NO
400:        .DC.L  _NO
401:        .DC.L  _NO
402:        .DC.L  _NO
403:        .DC.L  _NO
404:        .DC.L  _NO
405:        .DC.L  _NO
406:        .DC.L  _NO
407:        .DC.L  _NO
408:        .DC.L  _NO
409:        .DC.L  _NO
410:        .DC.L  _NO
411:        .DC.L  _NO
412:        .DC.L  _NO
413:        .DC.L  _NO
414:        .DC.L  _NO
415:        .DC.L  _NO
416:        .DC.L  _NO
417:        .DC.L  _NO
418:        .DC.L  _NO
419:        .DC.L  _NO
420:        .DC.L  _NO
421:        .DC.L  _NO
422:        .DC.L  _NO
423:        .DC.L  _NO
424:        .DC.L  _NO
425:        .DC.L  _CLMUL
426:        .DC.L  _CLDIV
427:        .DC.L  _CLMOD
428:        .DC.L  _CUMUL
429:        .DC.L  _CUDIV
430:        .DC.L  _CUMOD
431:        .DC.L  _CLTOD
432:        .DC.L  _CDTOL
433:        .DC.L  _CLTOF
434:        .DC.L  _CFTOL
435:        .DC.L  _CFTOD
436:        .DC.L  _CDTOF
437:        .DC.L  _CDCMP
438:        .DC.L  _CDADD
439:        .DC.L  _CDSUB
440:        .DC.L  _CDMUL
441:        .DC.L  _CDDIV
442:        .DC.L  _CDMOD
443:        .DC.L  _CFCMP
444:        .DC.L  _CFADD
445:        .DC.L  _CFSUB
446:        .DC.L  _CFMUL
447:        .DC.L  _CFDIV
448:        .DC.L  _CFMOD
449:        .DC.L  _CDTST
450:        .DC.L  _CFTST
451:        .DC.L  _CDINC
452:        .DC.L  _CFINC
453:        .DC.L  _CDDEC
454:        .DC.L  _CFDEC
455:        .DC.L  _FEVARG
456:        .DC.L  _FEVECS
457: _CFRET:
458:        MOVEM.L D0,(A6)
459: _CFRET0:
460:        MOVEM.L (SP)+,D0
461:        RTS
462: _CDRET:
463:        MOVEM.L D0-D1,(A6)
464: _CDRET0:
465:        MOVEM.L (SP)+,D0-D1
466:        RTS
467: _C4RET:
```

▶いま一度，万感の思いを込めて終了のベルが鳴る。いま一度，万感の思いを込めて，受験生が往く。さらば第一志望大学。さらば共通一次学力試験。出典：「さよなら銀河鉄道999」より。

東野　世士宏（19）奈良県

```
468:         MOVEM.L D0-D3,(A6)
469: _C4RETO:
470:         MOVEM.L (SP)+,D0-D3
471: _NO2:
472:         RTS
473: _D_S_RET:
474:         BSR    DBL_IEEE_SNG
475: _D_S_RETO:
476:         MOVEM.L (SP)+,D1
477:         RTS
478: *1
479: '倍精度絶対値
480: _DABS:
481:         BCLR   #31,D0
482:         MOVE   #0,CCR
483:         RTS
484: *1
485: _RND:
486:         MOVEM.L D2-D3,-(SP)
487:         BSR     GEN_RND
488:         BSR    LONG_IEEE_DBL
489:         MOVE.L #$40F00000,D2
490:         MOVEQ  #0,D3
491:         BSR    IE_SDDIV
492:         MOVEM.L (SP)+,D2-D3
493:         RTS
494:
495: *1 IEEE
496: _CFTST:
497:         MOVE.L D0,-(SP)          ;MOVEM.L
498:         MOVE.L (A6),D0           ;MOVEM.L
499:         PEA     _CFRETO(PC)
500: ***     BRA    _FTST
501: '単精度テスト
502: _FTST:
503:         TST.L  D0
504:         RTS
505: *1
506: '単精度反転
507: _FNEG:
508:         TST.L  D0
509:         BEQ.B  _FNEG_0
510:         BCHG   #31,D0
511:         TST.L  D0
512: _FNEG_0:
513:         RTS
514: *1
515: '単精度比較
516: _CFCMP:
517:         MOVEM.L D0-D1,-(SP)
518:         MOVEM.L (A6),D0-D1
519:         PEA     _CDRETO(PC)
520: ***     BRA    _FCMP
521: _FCMP:
522:         MOVEM.L D1-D4,-(SP)      ;MOVEM.L
523:         BSR    SNG_EX_DBL
524:         BSR    IE_SDCMP
525:         MOVEM.L (SP)+,D1-D4
526:         RTS
527: *1
528: '単精度加算
529: _CFADD:
530:         MOVEM.L D0-D1,-(SP)
531:         MOVEM.L (A6),D0-D1
532:         PEA     _CDRET(PC)
533: ***     BRA    _FADD
534: _FADD:
535:         MOVEM.L D1-D4,-(SP)      ;MOVEM.L
536:         BSR    SNG_2_DBL
537:         BSR    IE_SDADD
538:         BSR    DBL_IEEE_SNG
539:         MOVEM.L (SP)+,D1-D4
540:         RTS
541: *1
542: '単精度減算
543: _CFSUB:
544:         MOVEM.L D0-D1,-(SP)
545:         MOVEM.L (A6),D0-D1
546:         PEA     _CDRET(PC)
547: ***     BRA    _FSUB
548: _FSUB:
549:         MOVEM.L D1-D4,-(SP)      ;MOVEM.L
550:         BSR    SNG_EX_DBL
551:         BSR    IE_SDSUB
552:         BSR    DBL_IEEE_SNG
553:         MOVEM.L (SP)+,D1-D4
554:         RTS
555: *1
556: '単精度乗算
557: _CFMUL:
558:         MOVEM.L D0-D1,-(SP)
559:         MOVEM.L (A6),D0-D1
560:         PEA     _CDRET(PC)
561: ***     BRA    _FMUL
562: _FMUL:
563:         MOVEM.L D1-D4,-(SP)      ;MOVEM.L
564:         BSR    SNG_2_DBL
565:         BSR    IE_SDMUL
566:         BSR    DBL_IEEE_SNG
567:         MOVEM.L (SP)+,D1-D4
568:         RTS
569: *1
570: '単精度除算
571: _CFDIV:
572:         MOVEM.L D0-D1,-(SP)
573:         MOVEM.L (A6),D0-D1
574:         PEA     _CDRET(PC)
575: ***     BRA    _FDIV
576: _FDIV:
577:         MOVEM.L D1-D4,-(SP)      ;MOVEM.L
578:         BSR    SNG_EX_DBL
579:         BSR    IE_SDDIV
580:         BSR    DBL_IEEE_SNG
581:         MOVEM.L (SP)+,D1-D4
582:         RTS
583: *1
584: '単精度絶対値
585: _FABS:
586:         BCLR   #31,D0
587:         TST.L  D0
588:         RTS
589: *1
590: '単精度剰余
591: _CFMOD:
592:         MOVEM.L D0-D1,-(SP)
593:         MOVEM.L (A6),D0-D1
594:         PEA     _CDRET(PC)
595: ***     BRA    _FMOD
596: _FMOD:
597:         MOVEM.L D1-D4,-(SP)
598:         BSR    SNG_EX_DBL
599:         BSR    IE_SDMOD
600:         BSR    DBL_IEEE_SNG
601:         MOVEM.L (SP)+,D1-D4
602:         RTS
603: *1
604: _FPOWER:
605:         MOVEM.L D1-D4,-(SP)
606:         BSR    SNG_EX_DBL
607:         BSR    IE_SPOWER
608:         BSR    DBL_IEEE_SNG
609:         MOVEM.L (SP)+,D1-D4
610:         RTS
611: *1
612: _FRND:
613:         BSR     GEN_RND
614:         MOVEM.L D1-D3,-(SP)
615:         BSR    LONG_IEEE_DBL
616:         MOVE.L #$40F00000,D2
617:         MOVEQ  #0,D3
618:         BSR    IE_SDDIV
619:         BSR    DBL_IEEE_SNG
620:         MOVEM.L (SP)+,D1-D3
621:         RTS
622: *1 IEEE
623: _CFINC:
624:         MOVE.L D0,-(SP)          ;MOVEM.L
625:         MOVE.L (A6),D0           ;MOVEM.L
626:         PEA     _CFRET(PC)
627: ***     BRA    _FADDONE
628: _FADDONE:
629:         MOVE.L D1,-(SP)
630:         BSR    SNG_IEEE_DBL
631:         BSR    IE_SDADDONE
632:         BSR    DBL_IEEE_SNG
633:         MOVEM.L (SP)+,D1
634:         RTS
635: *1 IEEE
636: _CFDEC:
637:         MOVE.L D0,-(SP)          ;MOVEM.L
638:         MOVE.L (A6),D0           ;MOVEM.L
639:         PEA     _CFRET(PC)
640: *****   BRA    _FSUBONE
641: _FSUBONE:
642:         MOVE.L D1,-(SP)
643:         BSR    SNG_IEEE_DBL
644:         BSR    IE_SDSUBONE
645:         BSR    DBL_IEEE_SNG
646:         MOVEM.L (SP)+,D1
647:         RTS
648: *1
649: _IDIV:
650:         TST.L  D1
651:         BEQ.B   IDIV_ER          ;DIVED BY ZERO.
652:         MOVEM.L D2-D3,-(SP)
653:         BSR    SB_IDIV
654:         MOVE.L D2,D1
655:         MOVEM.L (SP)+,D2-D3
656:         RTS
657: '0除算エラー
658: _IDIV_ER:
659:         ORI.B  #$01,CCR          ;C=1
660:         RTS
661:
662: *1
663: '倍精度ＩＥＥＥを指数部と仮数部に分ける
664: _DFREXP:
665:         MOVE.L D0,D2
666:         BCLR   #31,D2            ;符号0
667:         OR.L   D1,D2
668:         BEQ.B  DFREXP_0          ;値０？
669:         MOVE.L D0,D2
670:         ANDI.L #$7FF00000,D2     ;IEEE指数部取り出し
671:         ROL.L  #8,D2
672:         ROL.L  #4,D2
673:         SUBI.L #$000003FF,D2     ;指数部
674:         ANDI.L #$800FFFFF,D0
675:         ORI.L  #$3FF00000,D0     ;仮数部のみ倍精度IEEEに
676:         RTS
677: *1
678: _DFREXP_0:
679:         MCLRL  D0                ;指数部も仮数部も０
680:         MCLRL  D1
681:         MCLRL  D2
682:         RTS
683: *1    i
684: 'Ｘ＊２
685: _DLDEXP:
686:         MOVE.L D2,-(SP)          ;MOVEM.L
687:         ADDI.L #$000003FF,D2
688:         MOVE.L D2,-(SP)          ;指数＋
689:         MOVE.L D0,D2
690:         BCLR   #31,D2            ;符号０
691:         OR.L   D1,D2
692:         BEQ.B  DLDEXP_0          ;Ｘ値０？
693:         MOVE.L D0,D2
694:         ANDI.L #$7FF00000,D2     ;IEEE指数部
695:         ROL.L  #8,D2
696:         ROL.L  #4,D2
697:         SUBI.L #$000003FF,D2
698:         ADD.L  (SP)+,D2
699:         CMP.L  #$00000800,D2     ;ＯＶＥＲチェック
700:         BCC.B  DLDEXP_OV
701:         ROR.L  #4,D2
702:         ROR.L  #8,D2
703:         ANDI.L #$800FFFFF,D0
704:         OR.L   D2,D0             ;仮数と符号に指数を合
成
705:         MOVEM.L (SP)+,D2
706:         RTS
707: '０のとき
708: _DLDEXP_0:
709:         ADDQ.L #4,SP
710:         MCLRL  D0                ;０を返す
711:         MCLRL  D1
712:         MOVEM.L (SP)+,D2
713:         RTS
714: '指数部オーバー
715: _DLDEXP_OV:
716:         ORI.B  #$01,CCR          ;CY=1
717:         MOVEM.L (SP)+,D2
718:         RTS
719: *1
720: '単精度ＩＥＥＥを指数部と仮数部に分ける
721: _FFREXP:
722:         MOVE.L D0,D1
723:         ANDI.L #$7FFFFFFF,D1     ;値０？
724:         BEQ.B  FFREXP_0
725:         ANDI.L #$7F800000,D1     ;IEEE指数
726:         ROL.L  #8,D1
727:         ROL.L  #1,D1
728:         SUBI.L #$0000007F,D1     ;指数部
729:         ANDI.L #$807FFFFF,D0
730:         ORI.L  #$3F800000,D0     ;仮数部
731:         RTS
732: '０のとき
733: _FFREXP_0:
734:         CLR.L  D0                ;０を返す
735:         RTS
736: *1    i
737: 'Ｘ＊２
738: _FLDEXP:
739:         MOVE.L D1,-(SP)          ;MOVEM.L
740:         ADDI.L #$0000007F,D1
741:         MOVE.L D1,-(SP)
742:         MOVE.L D0,D1
743:         ANDI.L #$7FFFFFFF,D1     ;値０？
744:         BEQ.B  FLDEXP_0
745:         ANDI.L #$7F800000,D1     ;IEEE指数
746:         ROL.L  #8,D1
747:         ROL.L  #1,D1
748:         SUBI.L #$0000007F,D1     ;指数部
749:         ADD.L  (SP)+,D1
750:         CMP.L  #$00000100,D1     ;指数部最大値チェック
751:         BCC.B  FLDEXP_OV
752:         ROR.L  #1,D1
753:         ROR.L  #8,D1
754:         ANDI.L #$807FFFFF,D0
755:         OR.L   D1,D0             ;指数部と仮数部合成
756:         MOVEM.L (SP)+,D1
757:         RTS
758: '０を返す
759: _FLDEXP_0:
760:         ADDQ.L #4,SP
761:         CLR.L  D0                ;０を返す
762:         MOVEM.L (SP)+,D1
763:         RTS
764: '指数部オーバー
765: _FLDEXP_OV:
766:         ORI.B  #$01,CCR          ;CY=1
767:         MOVEM.L (SP)+,D1
768:         RTS
769: *1
770: _FEVARG:
771:         MOVE.L #"IEEE",D0        ;$49454545,D0
772:         MOVE.L #"SOFT",D1        ;$534F4654,D1
773:         RTS
774:
775: M_COPYRIGHT:
776:         DC.B   $D,$A,"浮動小数点演算パッケージ"
777:         DC.B   "エミュレータ for X68000 version 1.00+",$D,$A
778:         DC.B   "（ＩＥＥＥフォーマット）",$D,$A
779:         DC.B   0,0,"M.YAMAGUCHI",0,0
780:         .EVEN
781: *1
782: _FFIX:
783:         MOVE.L D1,-(SP)          ;MOVEM.L
784:         BSR    SNG_IEEE_DBL
785:         PEA     D_S_RET(PC)
786: ***     BRA    IE_SDFIX
787: _DFIX:
788: ¥¥***   BRA    IE_SDFIX
789: '浮動少数の整数部を返す IEEE
790: IE_SDFIX:
791:         MOVEM.L D2-D3,-(SP)
792: IE_144:
793:         BSR.B  IE_DFIX_S         ;整数部取り出し
794:         MOVEM.L (SP)+,D2-D3
795:         RTS
796: '整数部取り出し
797: IE_DFIX_S:
798:         MOVE.L D0,D2
799:         SWAP   D2
800:         LSR.W  #4,D2
801:         ANDI.W #$7FF,D2          ;指数部を取り出す
802:         SUBI.W #$3FF,D2          ;２の補数にする
803:
804:         CMP.W  #$0015,D2
805:         BCS.B  IE_DFIX_LOW       ;２１ビットより小さい
806:         CMP.W  #$034,D2          ;仮数部は５２ビット
807:         BCS.B  IE_DFIX_HIG       ;２１ビットから５１ビ
ット
808:         TST.W  D2
809:         BMI.B  IE_147            ;１以下
810:         BRA.B  IE_R148           ;大きすぎる　→少数部は
ない
811: '２１ビットから５１ビットまで
812: IE_DFIX_HIG:
813:         SUBI.W #$0014,D2         ;-20BIT
814:         MOVEQ.L #0,D3
815:         MOVE   #$11,CCR          ;PATTERN
816:         ROXR.L D2,D3             ;仮数部位置合わせ
817:         NEG.L  D3
818:         AND.L  D3,D1             ;MASK LOW
819:         RTS
820: '０から２０ビットまで
821: IE_DFIX_LOW:
822:         MOVE.L #$100000,D3       ;PATTERN 20ﾋﾞｯﾄ以下
823:         MOVEQ  #0,D1             ;LOW=0
824:         LSR.L  D2,D3             ;PATTERN位置合わせ
825:         NEG.L  D3
826:         AND.L  D3,D0             ;MASK HIGH
827:         RTS
828:
829: '１以下で整数にならない
830: IE_147:
831:         MOVEQ  #0,D0  ;０を返す
832:         MOVEQ  #0,D1
833: IE_R148:
834:         RTS
835: *1
836: '等しいか大きい最大の数を返す
837: _FFLOOR:
838:         MOVE.L D1,-(SP)          ;MOVEM.L
839:         BSR    SNG_IEEE_DBL
840:         PEA     D_S_RET(PC)
841: ***     BRA    IE_SDFLOOR
842: _DFLOOR:
843: ¥¥**    BRA    IE_SDFLOOR
844: IE_SDFLOOR:
845:         MOVEM.L D2-D3,-(SP)
846:         TST.L  D0
847:         BPL    IE_144            ;整数部のみ
848:
849:         MOVE.L D0,D2
850:         MOVE.L D1,D3
851:         BSR    IE_SDFRAC
852:         BSR    IE_SDTST
853:         EXG.L  D2,D0
854:         EXG.L  D3,D1
855:         BEQ.B  IE_150            ;値０
856:
857:         BSR    IE_DFIX_S         ;整数部取り出し
858:         IE_REG 0,$03FF,D2,D3
859:         BSR    IE_SDSUB          ;－１
860: IE_150: MOVEM.L (SP)+,D2-D3
861:         RTS
862: *1
863: '等しいか大きい最小数を返す
864: _FCEIL:
865:         MOVE.L D1,-(SP)          ;MOVEM.L
866:         BSR    SNG_IEEE_DBL
867:         PEA     D_S_RET(PC)
868: ***     BRA    IE_SDCEIL
869: _DCEIL:
870: ¥¥***   BRA    IE_SDCEIL
871: IE_SDCEIL:
872:         MOVEM.L D2-D3,-(SP)
873:         TST.L  D0
874:         BMI    IE_144            ;整数部のみ
875:
876:         MOVE.L D0,D2
877:         MOVE.L D1,D3
878:         BSR.B  IE_SDFRAC
879:         BSR    IE_SDTST
880:         EXG.L  D2,D0
881:         EXG.L  D3,D1
882:         BEQ.B  IE_152            ;値０
883:
884:         MOVE.L D0,D2
885:         SWAP   D2
886:         ANDI.W #$7FF0,D2
887:         CMP.W  #$4330,D2
888:         BCC.B  IE_152            ;大きすぎる
889:
890:         BSR    IE_DFIX_S         ;整数取り出し
891:         IE_REG 0,$3FF,D2,D3      ;＋１
892:         BSR    IE_SDADD
893: IE_152: MOVEM.L (SP)+,D2-D3
894:         RTS
895:
896: *2
897: _FFRAC:
898:         MOVE.L D1,-(SP)
899:         BSR    SNG_IEEE_DBL
```

▶今年の共通一次はとんでもない。間違いは２カ所もあったし，平均点の調整もあったし。工学部の電子系志望者は大半が物理の選択者であるため，合格ボーダーは上がるけど，化学を選択した僕の点数は上がらない。いったいどうしてくれるんだよー。

坊農　誠（18）奈良県

```
900:        PEA     D_S_RET(PC)
901: ***     BRA    IE_SDFRAC
902: _DFRAC:
903: ****    BRA    IE_SDFRAC
904: ' IEEE FORM                  ;少数部のみの取り出し
?
905: IE_SDFRAC:
906:        MOVEM.L D2-D3,-(SP)
907:        MOVE.L  D0,D2
908:        SWAP    D2
909:        LSR.W   #4,D2
910:        ANDI.W  #$07FF,D2      ;指数部
911:        CMP.W   #$03FF,D2
912:         BCS.B  IE_157         ;小さい
913:        CMP.W   #$0433,D2
914:         BCC.B  IE_158         ;大きい 52BIT
915:
916:        MOVE.L  D0,D3
917:        ANDI.L  #$FFFFF,D0
918: IE_154:
919:        ADD.L   D1,D1          ;仮数位置合わせ＊２
920:        ADDX.L  D0,D0          ;
921:        SUBQ.W  #1,D2          ;指数÷２
922:        CMP.W   #$03FF,D2
923:         BCC.B  IE_154         ;LOOP
924:        BCLR    #20,D0
925:         BNE.B  IE_155
926:        AND.L   #$FFFFF,D0
927:        TST.L   D0
928:         BNE.B  IE_154         ;LOOP
929:        TST.L   D1
930:         BNE.B  IE_154         ;LOOP
931: IE_158:
932:        MCLRL   D0             ;値０
933:        MCLRL   D1
934:        MOVEM.L (SP)+,D2-D3
935:        RTS
936:
937: IE_155:
938:        AND.L   #$FFFFF,D0
939:        TST.L   D3
940:         BPL.B  IE_156         ;プラスはそのまま
941:        BSET    #31,D0
942: IE_156:
943:        LSL.W   #4,D2
944:        SWAP    D2
945:        CLR.W   D2
946:        OR.L    D2,D0          ;指数Ｄ２と仮数Ｄ０を
合成
947: IE_157:
948:        ANDI.B  #$FE,CCR       ;CY=0
949:        MOVEM.L (SP)+,D2-D3
950:        RTS
951:
952: *1
953: '整数－倍精度実数－変換 IEEE
954: __CLTOD:
955:        MOVEM.L D0-D1,-(SP)
956:        MOVEM.L (A6),D0-D1
957:        PEA     _CDRET(PC)
958: *****   BRA    __LTOD
959: __LTOD:
960: ******  BRA    LONG_IEEE_DBL
961: LONG_IEEE_DBL:
962:        TST.L   D0
963:         BEQ.B  IE_162         ;０のとき
964:        MOVE.L  D2,-(SP)
965:        MOVE.L  D0,-(SP)
966:        ANDI.L  #$80000000,(SP) ;符号のみ保存する
967:        MOVEQ   #0,D1          ;仮数部下位桁は０
968:        TST.L   D0             ;絶対値
969:         BPL.B  IE_160
970:        NEG.L   D0             ;マイナスの時はプラス
にする
971: IE_160:
972:        MOVE.L  #$41E0-$B0,D2  ;指数部
973:        CMP.L   #$200000,D0
974:         BCC    IE_160F
975:         BRA    IE_160E
976: IE_160L:
977:        SUBI.W  #$10,D2        ;指数－１
978:        ADD.L   D0,D0          ;仮数位置合わせ
979: IE_160E:
980:        BTST    #20,D0
981:         BEQ.B  IE_160L
982: IE_161:
983:        SWAP    D0
984:        ANDI.W  #$F,D0         ;指数部０
985:        OR.W    D2,D0          ;指数部合成
986:        AND.W   #$7FFF,D0      ;符号＋
987:        SWAP    D0
988:
989:        OR.L    (SP)+,D0       ;仮数の符号を合成する
990:        MOVE.L  (SP)+,D2
991:        RTS
992: '０のとき
993: IE_162:
994:        MOVEQ   #0,D1
995:        RTS
996: IE_160F:
997: IE_161FL:
998:        ADDI.W  #$10,D2        ;指数＋１
999:        LSR.L   #1,D0          ;仮数部右シフト
1000:       ROXR.L  #1,D1
1001:       CMP.L   #$200000,D0
1002:        BCC    IE_161FL
1003:       BRA     IE_161
1004: *1
1005: '倍精度－整数－変換 IEEE
1006: __CDTOL:
1007:       MOVEM.L D0-D1,-(SP)
1008:       MOVEM.L (A6),D0-D1
1009:       PEA     _CDRET(PC)
1010: *****  BRA    __DTOL
1011: __DTOL:
1012: ****   BRA    DBL_IEEE_LONG
1013: DBL_IEEE_LONG:
1014:       MOVE.L  D2,-(SP)
1015:
1016:       MOVE.L  D0,D2
1017:       SWAP    D2
1018:       LSR.W   #4,D2
1019:       ANDI.W  #$7FF,D2       ;指数部取りだし
1020:       SUBI.W  #$3FF,D2
1021:        BCS.B  IE_166         ;１以下の時０とする
1022:
1023:       CMP.W   #$01F,D2       ;指数３１ビット以上
1024:        BCC.B  IE_169         ;大きすぎる フラグを立
て戻る
1025:       MOVE.L  D0,-(SP)
1026:
1027:       SWAP    D0             ;仮数部取り出し
1028:       OR.W    #$10,D0        ;省略仮数１
1029:       AND.W   #$1F,D0
1030:       SWAP    D0
1031:
1032:       CMP.W   #$014,D2       ;指数２０ビット以上
1033:        BCC    IE_1640
1034: '１９ビット以下の時
1035: IE_164:
1036:       LSR.L   #1,D0
1037:       ADDQ    #1,D2
1038:       CMP     #20,D2
1039:        BCS    IE_164
1040: IE_164E:
1041:       TST.L   (SP)+
1042:        BPL.B  IE_P165
1043:       NEG.L   D0             ;絶対値
1044: IE_P165:
1045:       MOVE.L  (SP)+,D2
1046:       RTS
1047: '２０ビット以上の時
1048: IE_164L:
1049:       ADD.L   D1,D1          ;左シフト
1050:       ADDX.L  D0,D0
1051:       SUBQ.W  #1,D2
1052: IE_1640:
1053:       CMP.W   #20,D2
1054:        BNE    IE_164L
1055:       BRA     IE_164E
1056: '０にする
1057: IE_166:
1058:       CLR.L   D0
1059: IE_167:
1060:       MOVE.L  (SP)+,D2
1061:       RTS
1062: '３１ビット以上の時
1063: IE_169:
1064: **      BNE.B  IE_1691
1065:       MOVE.L  D0,D2
1066:       LSR.L   #1,D0
1067:       ROXR.L  #1,D1
1068:       LSR.L   #1,D0
1069:       ROXR.L  #1,D1
1070:       LSR.L   #1,D0
1071:       ROXR.L  #1,D1
1072:       LSR.L   #1,D0
1073:       ROXR.L  #1,D1
1074:       MOVE.W  D0,D1
1075:       SWAP    D1
1076:       MOVE.L  D1,D0
1077:       ADD.L   D2,D2
1078:       ROXR.L  #1,D0
1079: IE_1691:
1080:       MOVE.L  (SP)+,D2
1081:       MOVE    #$0003,CCR     ;CY=1 V=1
1082:       RTS
1083:
1084: *1
1085: '符号付整数乗算
1086: __CLMUL:
1087:       MOVEM.L D0-D1,-(SP)
1088:       MOVEM.L (A6),D0-D1
1089:       PEA     _CDRET(PC)
1090: *****  BRA    __LMUL
1091: __LMUL:
1092:       MOVEM.L D1-D2/D7,-(SP)
1093:       MOVE.L  D0,D7
1094:        BEQ.B  _LMUL_0        ;０
1095:        BPL.B  P171
1096:       NEG.L   D0             ;絶対値化
1097: P171:  EOR.L  D1,D7
1098:       TST.L   D1
1099:        BEQ.B  _LMUL_0        ;０
1100:        BPL.B  P172
1101:       NEG.L   D1             ;絶対値化
1102: P172:
1103:        BSR.B  SB_IMUL        ;乗算
1104:       TST.L   D0
1105:        BNE.B  _LMUL_ER
1106:       MOVE.L  D1,D0
1107:       TST.L   D7
1108:        BPL.B  _LMUL_PLUS     ;同符号
1109: 'マイナスにする
1110:       CMP.L   #$80000000,D0
1111:        BHI.B  _LMUL_ER       ;OVER
1112:       NEG.L   D0             ;反転 LONG INT
1113:       ANDI.B  #$FE,CCR       ;CY=0
1114:       MOVEM.L (SP)+,D1-D2/D7
1115:       RTS
1116: 'マイナスならオーバー
1117: _LMUL_PLUS:
1118:       TST.L   D0
1119:        BPL.B  P173
1120: _LMUL_ER:
1121:       MOVE    #$0003,CCR     ;CY=1 V=1
1122: P173:  MOVEM.L (SP)+,D1-D2/D7
1123:       RTS
1124: '０のとき
1125: _LMUL_0:
1126:       CLR.L   D0             ;値０
1127:       MOVEM.L (SP)+,D1-D2/D7
1128:       RTS
1129: *3
1130: '符号無し整数乗算
1131: __CUMUL:
1132:       MOVEM.L D0-D1,-(SP)
1133:       MOVEM.L (A6),D0-D1
1134:       PEA     _CDRET(PC)
1135: *****  BRA    __UMUL
1136: __UMUL:
1137:       MOVEM.L D1-D2,-(SP)
1138:       BSR.B   SB_IMUL
1139:       TST.L   D0
1140:        BNE.B  _UMUL_ER       ;OVER 32BIT
1141:       MOVE.L  D1,D0
1142:       MOVEM.L (SP)+,D1-D2
1143:       RTS
1144: _UMUL_ER:
1145:       MOVE    #3,CCR         ;CY=1 V=1
1146:       MOVEM.L (SP)+,D1-D2
1147:       RTS
1148: *1
1149: __IMUL:
1150:       MOVE.L  D2,-(SP)
1151:        BSR.B  SB_IMUL
1152:       MOVE.L  (SP)+,D2
1153:       RTS
1154: *3
1155: '３２ビット＊３２ビット－＞６４ビット
1156: SB_IMUL:
1157:       MOVE.W  D1,D2
1158:       MULU.W  D0,D2          ;D1.Low * D0.Low -> SP.LOW
1159:       MOVE.L  D2,-(SP)
1160:       CLR.L   -(SP)          ;SP.H = 0
1161:       MOVE.L  D0,D2
1162:       SWAP.W  D2
1163:       MULU.W  D1,D2          ;D1.Low * D2.High -> SP.MID
1164:       ADD.L   D2,$0002(SP)
1165:        BCC.B  J00180
1166:       ADDQ.W  #1,(SP)        ;CARY +1
1167: J00180:
1168:       SWAP.W  D1
1169:       MOVE.L  D0,D2
1170:       MULU.W  D1,D2          ;D1.High * D2.Low -> SP.MID
1171:
1172:       ADD.L   D2,$0002(SP)
1173:        BCC.B  J00181
1174:       ADDQ.W  #1,(SP)        ;CARY +1
1175: J00181: SWAP.W D0
1176:
1177:       MULU.W  D1,D0          ;D1.High * D0.High ->SP.HIGH
1178:       ADD.L   (SP)+,D0
1179:       MOVE.L  (SP)+,D1
1180:       RTS
1181: *2
1182: '符号付除算
1183: __CLDIV:
1184:       MOVEM.L D0-D1,-(SP)
1185:       MOVEM.L (A6),D0-D1
1186:       PEA     _CDRET(PC)
1187: ****   BRA    __LDIV
1188: __LDIV:
1189:       MOVEM.L D1-D3/D7,-(SP)
1190:       MOVE.L  D0,D7
1191:        BEQ.B  J00186         ;被除数０
1192:        BPL.B  P183
1193:       NEG.L   D0             ;絶対値化
1194: P183:  EOR.L  D1,D7
1195:       TST.L   D1
1196:        BEQ.B  _LDIV_0        ;除数０
1197:        BPL.B  P184
1198:       NEG.L   D1             ;絶対値化
1199: P184:   BSR   SB_IDIV
1200:       TST.L   D7
1201:        BPL.B  J00186
1202:       NEG.L   D0             ;マイナス
1203: J00185: ANDI.B #$FE,CCR     ;CY=0
1204:       MOVEM.L (SP)+,D1-D3/D7
1205:       RTS
1206:
1207: J00186: CMP.L  #$80000000,D0
1208:        BNE.B  J00185
1209:       SUBQ.L  #1,D0
1210:       MOVE    #$0003,CCR     ;CY=1 V=1
1211:       MOVEM.L (SP)+,D1-D3/D7
1212:       RTS
1213: '０で割るエラー
1214: _LDIV_0:
1215:       MOVEM.L (SP)+,D1-D3/D7
1216:       ORI.B   #$01,CCR       ;CY=1
1217:       RTS
1218: *1
1219: '剰余
1220: __CLMOD:
1221:       MOVEM.L D0-D1,-(SP)
1222:       MOVEM.L (A6),D0-D1
1223:       PEA     _CDRET(PC)
1224: ****   BRA    __LMOD
1225: __LMOD:
1226:       MOVEM.L D1-D3/D7,-(SP)
1227:       MOVE.L  D0,D7
1228:        BEQ.B  J00186         ;被除数０
1229:        BPL.B  P189
1230:       NEG.L   D0             ;絶対値化
1231: P189:  TST.L  D1
1232:        BEQ.B  _LDIV_0        ;除数０ エラー
1233:        BPL.B  P190
1234:       NEG.L   D1             ;絶対値化
1235: P190:   BSR.B SB_IDIV
1236:       TST.L   D7
1237:        BPL.B  P191
1238:       NEG.L   D2             ;符号反転
1239: P191:  MOVE.L D2,D0          ;剰余
1240:       MOVEM.L (SP)+,D1-D3/D7
1241:       RTS
1242:
1243: *1
1244: '符号無し除算
1245: __CUDIV:
1246:       MOVEM.L D0-D1,-(SP)
1247:       MOVEM.L (A6),D0-D1
1248:       PEA     _CDRET(PC)
1249: *****  BRA    __UDIV
1250: __UDIV:
1251:       MOVEM.L D1-D3/D7,-(SP)
1252:       TST.L   D0
1253:        BEQ.B  _UDIV_0        ;被除数０
1254:       TST.L   D1
1255:        BEQ.B  _LDIV_0        ;除数０  エラー
1256:        BSR.B  SB_IDIV
1257:       ANDI.B  #$FE,CCR       ;CY=0
1258: _UDIV_0:
1259:       MOVEM.L (SP)+,D1-D3/D7
1260:       RTS
1261: *5
1262: '３２ビット割る３２ビット
1263: SB_IDIV:
1264:       MCLRL   D2             ;
1265:       MOVEQ.L #$1F,D3        ;32 LOOP
1266: P195:
1267:       ADD.L   D0,D0          ;LSL.L #1,D0
1268:       ADDX.L  D2,D2          ;ROXL.L #1,D2
1269:       CMP.L   D1,D2
1270:        BCS.B  P196
1271:       ADDQ    #1,D0          ;BSET  #0,D0
1272:       SUB.L   D1,D2
1273: P196:
1274:       DBF     D3,P195
1275:       RTS
1276: *1 IEEE
1277: ' 倍精度テスト
1278: __CDTST:
1279:       MOVEM.L D0-D1,-(SP)
1280:       MOVEM.L (A6),D0-D1
1281:       PEA     _CDRETO(PC)
1282: ***    BRA    __DTST
1283: __DTST:
1284: ******  BRA   IE_SDTST
1285: IE_SDTST:
1286: S_DTST:
1287:       TST.L   D0             ;上位値テスト
1288:        BNE.B  DTST_NZ
1289:       TST.L   D1             ;下位値テスト
1290:       AND.W   #4,CCR         ;Z=X C=N=V=0
1291: DTST_NZ:
1292:       RTS
1293:
1294: *1
1295: '倍精度比較'比較チェック
1296: __CDCMP:
1297:       MOVEM.L D0-D3,-(SP)
1298:       MOVEM.L (A6),D0-D3
1299:       PEA     _C4RETO(PC)
1300: *****  BRA    __DCMP
1301: __DCMP:
1302: *****  BRA    IE_SDCMP
1303: IE_SDCMP:
1304:        BSR.B  IE_SDCMP1
1305:       ANDI.B  #$F5,CCR       ;C=X Z=X   N=V=0
1306:        BCC.B  IE_R200
1307:       ORI.B   #$08,CCR       ;N=1
1308: IE_R200:
1309:       RTS
1310: '比較ルーチン  ＣＹ，Ｚを返す
1311: IE_SDCMP1:
1312:       MOVE.L  D0,-(SP)
1313:       EOR.L   D2,(SP)+
1314:        BMI.B  IE_SDCMP2      ;一方はマイナス
1315:       TST.L   D0
1316:        BPL.B  IE_SDCMP3
1317:        BSR.B  IE_SDCMP3      ;両方マイナスのときの
比較後反転
1318:        BEQ.B  IE_R200
1319:       EORI.B  #$01,CCR       ;-CY
1320:       RTS
1321:
1322: '一方はマイナス
1323: IE_SDCMP2:
1324:       ROL.L   #1,D0
1325:       ROR.L   #1,D0          ;B32->CY
1326:       ANDI.B  #$FB,CCR       ;Z=0 N,V,C(Ｎは未使用)
1327:       RTS
1328:
1329: IE_SDCMP3:
```

▶「アーキテクチャからのマシン語入門」を読んだとたんに，これまでのマシン語入門記事で仕入れた知識の断片が，ひとつに収束したような感じです。これからマシン語に手を出してみようかという気になりました。

長井　勝人（22）三重県

```
1330:         CMP.L   D2,D0
1331:          BNE    IE_SDCMP39
1332:         CMP.L   D3,D1
1333: IE_SDCMP39:
1334:         RTS
1335:
1336:
1337: *1
1338: '倍精度反転
1339: _DNEG:
1340:         PEA     IE_SDTST(PC)
1341: ***      BRA    IE_SDNEG
1342: IE_SDNEG:
1343: S_DNEG:
1344:         TST.L   D1              ;D0.D1が０か？
1345:          BEQ.B  _DNEG_D0
1346:         BCHG    #31,D0
1347:         RTS
1348: _DNEG_D0:
1349:         MOVE.L  D0,-(SP)
1350:         ANDI.L  #$7FFFFFFF,(SP)+
1351:          BEQ.B  R_NEG
1352:         BCHG    #31,D0          ;符号反転
1353: R_NEG:
1354:         RTS
1355: *1 IEEE
1356: __CDINC:
1357:         MOVEM.L D0-D1,-(SP)
1358:         MOVEM.L (A6),D0-D1
1359:         PEA     _CDRET(PC)
1360: ***      BRA    __DADDONE
1361: _DADDONE:
1362: *****    BRA    IE_SDADDONE
1363: IE_SDADDONE:
1364:         MOVEM.L D2-D6,-(SP)
1365:         IE_REG  0,$3FF,D2,D3    ;1
1366:          BSR.B  IE_J00211
1367: IE_DADDRET:
1368:         MOVE    D6,CCR
1369:         MOVEM.L (SP)+,D2-D6
1370:         RTS
1371: '一方がマイナスのとき減算ルーチンへ
1372: IE_J00212:
1373:         TST.L   D0
1374:          BPL    IE_J00225       ;D2.D3がマイナスの時　減算
1375:         EXG.L   D0,D2
1376:         EXG.L   D1,D3
1377:          BRA    IE_J00225       ;D0.D1がマイナスの時　交換後減算
1378:
1379: IE_M213:
1380:         TST.L   D1
1381:          BNE.B  IE_J00214
1382:         MOVE.L  D2,D0           ;D0.D1は０ならD2.D3をそのまま使用する
1383:         MOVE.L  D3,D1
1384:         BCLR    #31,D0          ;AND.L  #$7FFFFFFF,D0  ;符号プラスに
1385:         RTS
1386: IE_M215:
1387:         TST.L   D3
1388:          BNE.B  IE_J00215
1389:         RTS                     ;D2.D3が０ならD0.D1をそのままかえす
1390: *1
1391: '倍精度加算'加算サブルーチン
1392: __CDADD:
1393:         MOVEM.L D0-D3,-(SP)
1394:         MOVEM.L (A6),D0-D3
1395:         PEA     _C4RET(PC)
1396: ***      BRA    __DADD
1397: _DADD:
1398: *****    BRA    IE_SDADD
1399: IE_SDADD:
1400:         MOVEM.L D2-D6,-(SP)
1401:         PEA     IE_DADDRET(PC)
1402: ***      BRA    IE_J00211              ;加算本体
1403: '加算ルーチン本体
1404: IE_J00211:
1405:         MOVE.L  D0,D4           ;まずフラグチェック
1406:         EOR.L   D2,D4
1407:          BMI.B  IE_J00212       ;一方がマイナスのとき
1408:         TST.L   D0
1409:          BPL.B  IE_J00213       ;両方プラスの時　単純加算
1410:         PEA     IE_SDNEG(PC)
1411: ***      BRA.B  IE_J00213       ;両方マイナスの時　加算後反転
1412: '両方＋か両方－
1413: IE_J00213:
1414:         MOVEQ   #0,D6           ;TEMP FLAG INITIAL 0
1415:         ANDI.L  #$7FFFFFFF,D0   ;D0.D1チェック
1416:          BEQ.B  IE_M213
1417: IE_J00214:
1418:         ANDI.L  #$7FFFFFFF,D2   ;CHECK D2.D3
1419:          BEQ.B  IE_M215
1420: IE_J00215:
1421:
1422:         MOVE.L  D0,D4
1423:         SWAP    D4
1424:         ASR.W   #4,D4           ;指数部１
1425:         MOVE.L  D2,D5
1426:         SWAP    D5
1427:         ASR.W   #4,D5           ;指数部２
1428:
1429:         CMP.W   D5,D4           ;指数部を比較する
1430:          BCC.B  IE_P216
1431:         EXG.L   D0,D2           ;指数２＞指数１のとき交換する
1432:         EXG.L   D1,D3           ;D0.D1+D4 <—> D2.D3+D5
1433:         EXG.L   D4,D5
1434: IE_P216:
1435:         SUB.W   D5,D4           ;指数部の差
1436:         ADD.W   D4,D5           ;Ｄ５は大きい方の指数部
1437:
1438:         CMP.W   #$0035,D4       ;仮数部が５３ビット以上はエラー（仮数部は５２ビットだから）
1439:          BCC.B  IE_R220
1440:
1441:         SWAP    D0
1442:         SWAP    D2
1443:         OR.W    #$10,D0         ;仮数ビット１
1444:         OR.W    #$10,D2
1445:         AND.W   #$1F,D0         ;仮数部のみにする
1446:         AND.W   #$1F,D2
1447:         SWAP    D0
1448:         SWAP    D2
1449:
1450:         CMP.W   #32,D4          ;指数の差が３２ビット以上のときのシフト
1451:          BCS.B  IE_J00218
1452:
1453:         ADD.L   D3,D3
1454:         MOVE    SR,-(SP)
1455:         MOVE.L  D2,D3           ;３２ビット分シフト
1456:         MOVEQ.L #0,D2
1457:         MOVE    (SP)+,CCR
1458:         ANDI.W  #$1F,D4
1459:          BRA.B  IE_J00218
1460: IE_P217:
1461:         LSR.L   #1,D2           ;仮数部位置合わせ
1462:         ROXR.L  #1,D3
1463: IE_J00218:
1464:         DBF     D4,IE_P217
1465:         ADDX.L  D3,D1           ;仮数部加算：下位桁四捨五入
1466:         ADDX.L  D2,D0
1467:         BTST    #21,D0          ;仮数あふれ
1468:          BEQ.B  IE_P219
1469:         LSR.L   #1,D0           ;正規化処理
1470:         ROXR.L  #1,D1
1471:         ADDQ.W  #1,D5           ;指数＋１
1472: IE_P219:
1473:         LSL.W   #4,D5
1474:          BMI    IE_J00238
1475:         SWAP    D0
1476:         ANDI.W  #$F,D0
1477:         OR.W    D5,D0           ;指数合成
1478:         SWAP    D0
1479: IE_R220:
1480:         RTS
1481: *1 IEEE
1482: __CDDEC:
1483:         MOVEM.L D0-D1,-(SP)
1484:         MOVEM.L (A6),D0-D1
1485:         PEA     _CDRET(PC)
1486: ***      BRA    __DSUBONE
1487: *2
1488: _DSUBONE:
1489: *****    BRA    IE_SDSUBONE
1490: *3
1491: IE_SDSUBONE:
1492:         MOVEM.L D2-D6,-(SP)
1493:         MOVE.L  #$3FF00000,D2
1494:         MOVEQ.L #$00,D3
1495:          BSR.B  IE_J00223
1496: IE_DSUBRET:
1497:         MOVE    D6,CCR
1498:         MOVEM.L (SP)+,D2-D6
1499:         RTS
1500: '一方がマイナスのとき
1501: IE_J00224:
1502:         TST.L   D0
1503:          BPL    IE_J00213       ;加算
1504:          BSR    IE_J00213       ;加算後反転
1505:          BRA    IE_SDNEG
1506: IE_M226:
1507:         TST.L   D1
1508:          BNE.B  IE_J00226
1509:         MOVE.L  D2,D0           ;D0,D1が０のときはD2.D3を使用する
1510:         MOVE.L  D3,D1
1511:          BCLR   #31,D0          ;符号＋
1512:          BRA    IE_SDNEG        ;反転
1513: IE_M227:
1514:         TST.L   D3
1515:          BNE.B  IE_J00227
1516:         RTS                     ;両方とも０なら０
1517: IE_M229:
1518:         CMP.L   D3,D1
1519:          BNE.B  IE_J00229
1520:         MOVEQ.L #0,D0           ;同一のとき０
1521:         MOVEQ.L #0,D1
1522: IE_R228:
1523:         RTS
1524: *1
1525: '倍精度減算'減算サブルーチン
1526: __CDSUB:
1527:         MOVEM.L D0-D3,-(SP)
1528:         MOVEM.L (A6),D0-D3
1529:         PEA     _C4RET(PC)
1530: *****    BRA    __DSUB
1531: _DSUB:
1532: *****    BRA    IE_SDSUB
1533: IE_SDSUB:
1534:         MOVEM.L D2-D6,-(SP)
1535:         PEA     IE_DSUBRET(PC)
1536: ***      BRA.B  IE_J00223
1537: '減算ルーチン本体
1538: IE_J00223:
1539:         MOVE.L  D0,D4
1540:         EOR.L   D2,D4
1541:          BMI.B  IE_J00224       ;一方がマイナス
1542:         TST.L   D0
1543:          BPL.B  IE_J00225       ;両方プラス
1544:         EXG.L   D0,D2           ;マイナスのとき交換
1545:         EXG.L   D1,D3
1546: IE_J00225:
1547:         MCLRL   D6              ;FLAG INITIAL 0
1548:         ANDI.L  #$7FFFFFFF,D0   ;フラグを除いた値
1549:          BEQ.B  IE_M226
1550: IE_J00226:
1551:         ANDI.L  #$7FFFFFFF,D2   ;フラグを除いた値
1552:          BEQ.B  IE_M227
1553: IE_J00227:
1554:         CMP.L   D2,D0           ;比較
1555:          BEQ.B  IE_M229         ;上位桁一致
1556: IE_J00229:
1557:          BCC.B  IE_J00230       ;D0.D1>=D2.D3のとき
1558:         EXG.L   D0,D2           ;D0.D1<D2.D3のとき交換
1559:         EXG.L   D1,D3
1560:         PEA     IE_SDNEG(PC)
1561: ***      BRA    IE_J00230       ;減算処理後反転
1562: IE_J00230:
1563:         MOVE.L  D0,D4
1564:         SWAP    D4
1565:         ASR.W   #4,D4           ;指数部１
1566:
1567:         MOVE.L  D2,D5
1568:         SWAP    D5
1569:         ASR.W   #4,D5           ;指数部２
1570:
1571:         SUB.W   D5,D4           ;Ｄ４は指数の差
1572:         ADD.W   D4,D5           ;Ｄ５は大きい方の値を保持する
1573:
1574:         CMP.W   #$0035,D4       ;差が５３ビット以上は無効
1575:          BCC.B  IE_R228         ;236
1576:
1577:         SWAP    D0              ;ここでは仮数部上位を処理する
1578:         SWAP    D2
1579:         OR.W    #$10,D0         ;仮数１の省略ビットを立てる
1580:         OR.W    #$10,D2         ;仮数２の省略ビットを立てる
1581:         AND.W   #$1F,D0         ;仮数のみ取り出し
1582:         AND.W   #$1F,D2
1583:         SWAP    D0
1584:         SWAP    D2
1585:
1586:         MOVE.L  D7,-(SP)
1587:         MOVEQ.L #0,D7           ;Ｄ３よりさらに下位の桁をＤ７に設定する
1588:         CMP.W   #32,D4          ;指数の差が３２ビット以下の時シフト
1589:          BCS.B  IE_J00232
1590:
1591:         MOVE.L  D3,D7           ;D2.D3.D7を右に３２ビットシフトする
1592:         SWAP    D7
1593:         MOVE.L  D2,D3           ;指数の差が３２ビット以上のときのシフト
1594:         MOVEQ.L #0,D2
1595:
1596:         ANDI.W  #$1F,D4         ;指数部から３２を引く
1597:          BRA.B  IE_J00232
1598: IE_P231:
1599:         LSR.L   #1,D2           ;D2.D3.D7を右に１ビットシフトする
1600:         ROXR.L  #1,D3           ;仮数部位置合わせ
1601:         ROXR.W  #1,D7
1602: IE_J00232:
1603:         DBF     D4,IE_P231      ;必要なだけシフトする
1604:
1605:         AND.W   #$FFE0,D7       ;互換性のため下位ビットを削る
1606:         NEG.W   D7              ;下位桁より演算
1607:         SUBX.L  D3,D1           ;仮数部減算処理
1608:         SUBX.L  D2,D0           ;(D0.D1.D0) - (D2.D3.D7)
1609:          BRA.B  IE_J00234
1610:
1611: IE_P233:
1612:         SUBQ.W  #1,D5           ;演算後の正規化処理
1613:         ADD.W   D7,D7           ;D0.D1.D7 を左に１ビットシフトする
1614:         ADDX.L  D1,D1
1615:         ADDX.L  D0,D0
1616: IE_J00234:
1617:         BTST    #20,D0          ;仮数部の１の位置
1618:          BEQ.B  IE_P233
1619:
1620:         MOVEQ.L #0,D2
1621:         ADD.W   D7,D7           ;下位桁四捨五入
1622:          BCC.B  IE_235
1623:         ADDX.L  D2,D1           ;桁上がり
1624:         ADDX.L  D2,D0
1625:         BTST    #21,D0          ;あふれチェック
1626:          BEQ.B  IE_235
1627:         LSR.L   #1,D0           ;再び１ビット右シフト
1628:         ROXR.L  #1,D1           ;正規化処理２
1629:         ADDQ.W  #1,D5
1630: IE_235:
1631:         MOVE.L  (SP)+,D7
1632:         CMP.W   #$0800,D5       ;８００以上かマイナス
1633:          BCC.B  IE_M237         ;指数部オーバー
1634:
1635:         LSL.W   #4,D5           ;指数部桁合わせ
1636:         SWAP    D0
1637:         AND.W   #$F,D0          ;指数部０カット
1638:         OR.W    D5,D0           ;指数部仮数部合成
1639:         SWAP    D0
1640: IE_R236:
1641:         RTS
1642: IE_M237:
1643:         TST.W   D5
1644:          BPL.B  IE_J00238
1645: '指数部０以下
1646: '仮数０の時
1647: IE_J00237:
1648:         MCLRL   D0              ;RETURN 0
1649:         MCLRL   D1
1650:         MOVEQ.L #$01,D6
1651:         RTS
1652: '指数が大きすぎる時
1653: IE_J00238:
1654:         IE_REG  $7FFFFFFF,$FFFFFFFF,D0,D1
1655:         MOVEQ.L #$03,D6         ;CY=1 V=1
1656:         RTS
1657:
1658: IE_DMULRET:
1659:         MOVE    D6,CCR
1660:         MOVEM.L (SP)+,D2-D7/A0
1661:         RTS
1662:
1663: IE_M243:
1664:         TST.L   D3
1665:          BNE.B  IE_J00244
1666:         MOVEQ   #0,D0           ;両方とも０
1667:         MOVEQ   #0,D1
1668:         MOVEQ   #0,D6
1669:         RTS
1670: IE_M242:
1671:         TST.L   D1
1672:          BNE.B  IE_J00242
1673:         MOVEQ   #0,D6
1674:         RTS                     ;０なら０
1675: IE_244P0:
1676:         TST.L   D2
1677:          BNE.B  IE_244P9
1678:         TST.L   D3
1679:          BNE.B  IE_244P9
1680:         BRA     IE_MMERGE
1681: *1
1682: __CDMUL:
1683:         MOVEM.L D0-D3,-(SP)
1684:         MOVEM.L (A6),D0-D3
1685:         PEA     _C4RET(PC)
1686: *****    BRA    __DMUL
1687: '倍精度乗算
1688: _DMUL:
1689: ****     BRA    IE_SDMUL
1690: *22
1691: '倍精度乗算IEEE
1692: IE_SDMUL:
1693:         MOVEM.L D2-D7/A0,-(SP)
1694:         PEA     IE_DMULRET(PC)
1695: ***      BRA.B  IE_J00240
1696: *1
1697: '符号乗算
1698: IE_J00240:
1699:         MOVE.L  D0,D4
1700:         EOR.L   D2,D4           ;CHECK FLAG
1701:          BPL.B  IE_J00241
1702:         PEA     IE_SDNEG(PC)
1703: ***      BRA.B  IE_J00241       ;乗算後反転
1704: '本体
1705: IE_J00241:
1706:         ANDI.L  #$7FFFFFFF,D0   ;フラグを除いた仮数部チェック
1707:          BEQ.B  IE_M242
1708: IE_J00242:
1709:         ANDI.L  #$7FFFFFFF,D2   ;乗数もチェック
1710:          BEQ.B  IE_M243
1711: IE_J00244:
1712:         SWAP    D0
1713:         SWAP    D2
1714:         MOVE.W  D0,D7
1715:         MOVE.W  D2,D6
1716:         LSR.W   #4,D7           ;指数
1717:         LSR.W   #4,D6
1718:         SUBI.W  #$03FF*2,D7     ;２の補数で表現
1719:         ADD.W   D6,D7
1720:
1721:         AND.W   #$0F,D0         ;仮数のみ
1722:         OR.W    #$10,D0         ;仮数１
1723:         SWAP    D0
1724:         AND.W   #$0F,D2
1725:          BEQ.B  IE_244P0
1726: IE_244P9:
1727:         OR.W    #$10,D2
1728:         SWAP    D2
1729:
1730:         LINK    A6,#-$0018      ;３２ＢＹＴＥＳ確保
1731:          BSR.B  IE_SMUL128      ;仮数部乗算
1732:
1733:         MOVE.L  -$0E(A6),D0     ;D0.D1乗算結果
1734:         MOVE.L  -$0A(A6),D1
1735:         UNLK    A6              ;スタック開放
1736:
1737:         MOVEQ   #0,D2
1738:
```

▶シャープのノートワープロを半年ほど使用してきましたが，なかなか使いやすくて満足しています。特に年末に発売となった通信セットはたいへん重宝しています。なんといっても，会社の電話でパソコン通信ができるので，電話代のことで嫁さんに叱られることもない。

伊佐治　進（31）愛知県

```
1739:         LSR.L   #1,D0        ;３ビット右シフト
1740:         ROXR.L  #1,D1
1741:         LSR.L   #1,D0
1742:         ROXR.L  #1,D1
1743:         LSR.L   #1,D0
1744:         ROXR.L  #1,D1
1745:
1746:         BTST    #21+1,D0          ;演算結果の仮数を
チェックする
1747:          BEQ    IE_J00245
1748:         ADDQ.L  #2,D1        ;四捨五入
1749:         ADDX.L  D2,D0        ;桁上がり
1750:         LSR.L   #1,D0
1751:         ROXR.L  #1,D1
1752:          BRA    IE_245
1753: IE_J00245:
1754:         ADDQ.L  #1,D1        ;四捨五入
1755:         ADDX.L  D2,D0        ;桁上がり
1756:         LSR.L   #1,D0
1757:         ROXR.L  #1,D1
1758: IE_245:
1759:         BTST    #21,D0
1760:          BEQ    IE_NMERGE
1761:         LSR.L   #1,D0        ;正規化する
1762:         ROXR.L  #1,D1
1763:         ADDQ.W  #1,D7        ;指数に１を加える
1764:          BRA    IE_NMERGE
1765:
1766: '仮数部乗算
1767: IE_SMUL128:
1768: S_MUL128:
1769:         MOVEQ.L #0,D4
1770:         MOVE.L  A6,A0
1771:         MOVE.L  D4,-(A0)     ;浮動少数レジスターを
設定 128BIT  全て０
1772:         MOVE.L  D4,-(A0)
1773:         MOVE.L  D4,-(A0)
1774:         MOVEM.L D2-D4,-(A0)
1775:         TST.L   D1           ;下位桁０の時 SHORT PASS
1776:          BEQ.B  P_MUL_LOW
1777:         LEA.L   -$000A(A6),A0 ;格納アドレス ;乗数の
下位から順に掛ける
1778:         EXG.L   D0,D1
1779:          BSR.B  MUL_64       ;第１番目
1780:         SUBQ.W  #2,A0        ;上位桁へ移動
1781:         SWAP    D0
1782:          BSR.B  MUL_64       ;第２番目
1783:         EXG.L   D0,D1
1784: P_MUL_LOW:
1785:         LEA.L   -$000E(A6),A0
1786:          BSR.B  MUL_64       ;第３番目
1787:         SUBQ.W  #2,A0
1788:         SWAP    D0
1789: '        BRA.B  MUL_64       ;第４番目
1790: '１６ビット＊６４ビット分
1791: MUL_64:
1792:         MOVEM.W -$18(A6),D2-D5
1793:         MULU.W  D0,D5
1794:         MULU.W  D0,D3
1795:         MULU.W  D0,D4
1796:         MULU.W  D0,D2
1797:         ADD.L   D5,6(A0)     ;下位
1798:         MOVE.L  2(A0),D6
1799:         ADDX.L  D3,D6        ;上位桁＋キャリー
1800:          BCC.B  P247
1801:         ADDQ.W  #1,(A0)      ;桁上がり処理
1802: P247:
1803:         MOVE.L  D6,2(A0)     ;上位
1804:
1805:         ADD.L   D4,4(A0)     ;下位
1806:         MOVE.L  (A0),D6
1807:         ADDX.L  D2,D6        ;上位桁＋キャリー
1808:         MOVE.L  D6,(A0)      ;上位
1809:         RTS
1810:
1811: '1
1812: __FDIVTWO:
1813:         MOVE.L  D1,-(SP)          ;MOVEM.L
1814:          BSR    SNG_IEEE_DBL
1815:         PEA     _D_S_RET(PC)
1816: '''      BRA    IE_SDDIVTWO
1817: __DDIVTWO:
1818: ''''''   BRA    IE_SDDIVTWO
1819: ' ２で割る IEEE
1820: IE_SDDIVTWO:
1821:         BSR     IE_SDTST
1822:          BEQ.B  IE_R253      ;被除数０
1823:         MOVE.L  D0,-(SP)
1824:         ANDI.L  #$7FF00000,D0 ;指数部のみ
1825:          BEQ.B  IE_254
1826:         MOVE.L  (SP)+,D0
1827:         SUBI.L  #$100000,D0  ;実行
1828: IE_R253:
1829:         RTS
1830: '小さすぎる
1831: IE_254:
1832:         MOVE.L  (SP)+,D0
1833:         MOVE    #$0001,CCR       ;CY=1
1834:         RTS
1835: IE_DDIVRET:
1836:         MOVE    D6,CCR
1837:         MOVEM.L (SP)+,D2-D7
1838:         RTS
1839: IE_M258:
1840:         TST.L   D1
1841:          BNE.B  IE_259
1842:         CLR.L   D6           ;両方とも０のとき
1843:         RTS
1844: IE_M257:
1845:         TST.L   D3
1846:          BNE.B  IE_258
1847:          BRA    IE_267       ;除数０
1848: IE_259P0:
1849:         TST.L   D2
1850:          BNE.B  IE_259P9
1851:         TST.L   D3
1852:          BNE.B  IE_259P9
1853:          BRA    IE_MMERGE
1854: '1
1855: __CDDIV:
1856:         MOVEM.L D0-D3,-(SP)
1857:         MOVEM.L (A6),D0-D3
1858:         PEA     _C4RET(PC)
1859: '''''    BRA    _DDIV
1860: '倍精度除算
1861: __DDIV:
1862: ''''''   BRA    IE_SDDIV
1863: '除算サブルーチンIEEE
1864: IE_SDDIV:
1865:         MOVEM.L D2-D7,-(SP)
1866:         PEA     IE_DDIVRET(PC)
1867: '''      BRA.B  IE_256
1868: '1
1869: IE_256:
1870:         MOVE.L  D0,D4
1871:         EOR.L   D2,D4        ;フラグ
1872:          BPL.B  IE_257       ;除算
1873:         PEA     IE_SDNEG(PC)
1874: '''      BRA.B  IE_257       ;除算後反転
1875: IE_257:
1876:         ANDI.L  #$7FFFFFFF,D2 ;フラグを除いた仮数部
テスト
1877:          BEQ.B  IE_M257
1878: IE_258: ANDI.L  #$7FFFFFFF,D0
1879:          BEQ.B  IE_M258
1880: '除算
1881: IE_259:
1882:         SWAP    D0
1883:         SWAP    D2
1884:         MOVE.W  D0,D7        ;指数部分離
1885:         MOVE.W  D2,D6
1886:         LSR.W   #4,D7        ;指数
1887:         LSR.W   #4,D6
1888:         SUB.W   D6,D7        ;指数  差
1889:         AND.W   #$0F,D0      ;仮数のみ
1890:         OR.W    #$10,D0      ;仮数１
1891:         SWAP    D0
1892:         AND.W   #$0F,D2
1893:          BEQ    IE_259P0     :被除数が２の倍数なら…
1894: IE_259P9:
1895:         OR.W    #$10,D2
1896:         SWAP    D2
1897:         MOVEQ.L #0,D4
1898:         CMP.L   D2,D0        ;仮数比較
1899:          BNE.B  IE_P259_0
1900:         CMP.L   D3,D1        ;仮数下位比較
1901: IE_P259_0:
1902:          BCC.B  IE_260       ;D0.D1>D2.D3
1903:         SUBQ.W  #1,D7        ;D0.D1<D2.D3 指数－１
1904:         MOVEQ.L #0,D5
1905:          BRA.B  IE_261
1906: IE_260:
1907:         MOVEQ.L #$01,D5
1908:         SUB.L   D3,D1        ;D0.D1-D2.D3  仮数減算
1909:         SUBX.L  D2,D0
1910: '仮数部除算ループ
1911: IE_261:
1912:         ADD.L   D5,D5        ;仮数割る２
1913:          BCS.B  IE_262E
1914:         ADD.L   D1,D1        ;被除数D0.D1.D4.D5×２
1915:         ADDX.L  D0,D0        ;
1916:         CMP.L   D2,D0        ;仮数比較 D0.D1-D2.D3
1917:          BCS.B  IE_261
1918:          BNE.B  IE_P261
1919:         CMP.L   D3,D1
1920:          BCS.B  IE_261       ;ＰＡＳＳ
1921: IE_P261:
1922:         SUB.L   D3,D1        ;仮数減算 D0.D1-D2.D3
1923:         SUBX.L  D2,D0
1924:         ADDQ    #1,D5        ;商ビットをたてる
1925:          BRA.B  IE_261
1926: ''''
1927: IE_H262:
1928:         BTST    #20,D4
1929:          BNE.B  IE_MERGE0
1930: IE_H261:
1931:         ADD.L   D5,D5        ;仮数割る２
1932: IE_262E:
1933:         ADDX.L  D4,D4        ;被除数D0.D1 D4.D5×２
1934:         ADD.L   D1,D1        ;
1935:         ADDX.L  D0,D0        ;
1936:         CMP.L   D2,D0        ;仮数比較 D0.D1-D2.D3
1937:          BCS.B  IE_H262
1938:          BNE.B  IE_HP261_S
1939:         CMP.L   D3,D1
1940:          BCS.B  IE_H262      ;ＰＡＳＳ
1941: IE_HP261_S:
1942:         SUB.L   D3,D1        ;仮数減算 D0.D1-D2.D3
1943:         SUBX.L  D2,D0
1944:         ADDQ    #1,D5        ;商ビットをたてる
1945:         BTST    #20,D4       ;TST.L  D4
1946:          BEQ.B  IE_H261
1947: IE_MERGE0:
1948:         ADD.L   D1,D1        ;四捨五入処理
1949:         ADDX.L  D0,D0
1950:         CMP.L   D2,D0
1951:          BCS.B  IE_MERGE
1952:          BNE.B  IE_UP
1953:         CMP.L   D3,D1
1954:          BCS.B  IE_MERGE
1955: IE_UP:
1956:         MOVEQ   #0,D0        ;桁上がり
1957:         ADDQ.L  #1,D5
1958:         ADDX.L  D0,D4
1959:         BTST    #21,D4
1960:          BEQ.B  IE_MERGE
1961:         LSR.L   #1,D4        ;正規化する
1962:         ROXR.L  #1,D5
1963:         ADDQ.W  #1,D7
1964:
1965: '合成処理IEEE
1966: IE_MERGE:
1967:         MOVE.L  D4,D0        ;答え
1968:         MOVE.L  D5,D1
1969: IE_NMERGE:
1970:         MCLRL   D6           ;
1971:         ADDI.W  #$03FF,D7    ;指数部処理  げたばき
1972:          BEQ.B  IE_M265
1973:         CMP.W   #$0800,D7
1974:          BCC    IE_MEROV     ;指数が大きすぎる時
1975: IE_266:
1976:         SWAP    D0
1977:         ANDI.W  #$F,D0
1978:         LSL.W   #4,D7
1979:         OR.W    D7,D0        ;指数部合成
1980:         SWAP    D0
1981:         RTS
1982: IE_MEROV:
1983:         TST.W   D7
1984:          BMI    IE_J00237
1985:         BRA     IE_J00238
1986: IE_M265:
1987:         ANDI.L  #$FFFFF,D4   ;指数０。  仮数が０か
どうかチェックする
1988:          BNE.B  IE_266
1989:         TST.L   D5
1990:          BNE.B  IE_266
1991:          BRA    IE_J00237    ;仮数全て０の時
1992: '除数０の時
1993: IE_267:
1994:         MOVE.L  #$7FFFFFFF,D0 ;最大値を返す
1995:         MOVEQ.L #$FF,D1
1996:         MOVEQ.L #$05,D6      ;CY=1 Z=1
1997:         RTS
1998:
1999: CHK_ASCII:
2000:         BSR.B   CHK_LOWER    ;61-7A... CY=0  小文字アスキ
ー
2001:          BCC.B  R273
2002: '大文字チェック
2003: CHK_UPPER:
2004:         CMP.B   #$41,D0      ;41-5A... CY=0  大文字アスキ
ー
2005:          BCS.B  R273
2006:         CMP.B   #$5B,D0
2007:         EORI.B  #$01,CCR     ;-CY
2008: R273:   RTS
2009: '小文字チェック
2010: CHK_LOWER:
2011:         CMP.B   #$61,D0      ;61-7A ...CY=0  小文字アスキ
ー
2012:          BCS.B  R275
2013:         CMP.B   #$7B,D0
2014:         EORI.B  #$01,CCR     ;-CY
2015: R275:   RTS
2016: '１０進数チェック
2017: CHK_10NUM:
2018:         CMP.B   #$30,D0      ;30-39 ...CY=0 数字
2019:          BCS.B  R277
2020:         CMP.B   #$3A,D0
2021:         EORI.B  #$01,CCR     ;-CY
2022: R277:   RTS
2023: '２進数チェック
2024: CHK_2NUM:
2025:         CMP.B   #$30,D0      ;30-31 ...CY=0 ２進数字
2026:          BCS.B  R279
2027:         CMP.B   #$32,D0
2028:         EORI.B  #$01,CCR     ;-CY
2029: R279:   RTS
2030: '８進数チェック
2031: CHK_8NUM:
2032:         CMP.B   #$30,D0      ;30-37 ...CY=0 ８進数字
2033:          BCS.B  R281
2034:         CMP.B   #$38,D0
2035:         EORI.B  #$01,CCR     ;-CY
2036: R281:   RTS
2037: '１６進数チェック
2038: CHK_HEX:
2039:          BSR.B  CHK_10NUM    ;0-9
2040:          BCC.B  R284
2041:         CMP.B   #$41,D0      ;"A"
2042:          BCS.B  R284
2043:         CMP.B   #$47,D0      ;"G"
2044:          BCS.B  J00283
2045:         CMP.B   #$61,D0      ;"a"
2046:          BCS.B  R284
2047:         CMP.B   #$67,D0      ;"g"
2048: J00283: EORI.B  #$01,CCR     ;-CY
2049: R284:   RTS
2050:
2051: '小文字なら大文字へ変換
2052: TO_UPPER:
2053:          BSR.B  CHK_LOWER
2054:          BCS.B  R286
2055:         SUB.B   #$20,D0
2056: R286:   RTS
2057:
2058: '大文字なら小文字に変換
2059: TO_LOWER:
2060:          BSR.B  CHK_UPPER
2061:          BCS.B  R288
2062:         ADD.B   #$20,D0
2063: R288:   RTS
2064: '１６進数を数値に変換
2065: TO_HEXVAL:
2066:          BSR.B  TO_LOWER
2067:         CMP.B   #$61,D0      ;'a'
2068:          BCS.B  P290
2069:         ADD.B   #-$27,D0     ;OFFSET
2070: P290:   ADD.B   #-$30,D0     ;0-F
2071:         RTS
2072: '1
2073: __FTOS:
2074:         MOVE.L  D1,-(SP)     ;MOVEM.L
2075:          BSR    SNG_IEEE_DBL
2076:         PEA     _D_S_RET(PC)
2077: '''      BRA    _DTOS
2078: __DTOS:
2079: ''''''   BRA    _DTOS        ;
2080: '倍精度を文字列に変換
2081: _DTOS:
2082:         MOVEQ.L #$0E,D2
2083:         MOVE.L  A0,-(SP)
2084:          BSR    S_GCVT
2085:         MOVE.L  (SP),A0
2086:          BSR.B  J00292
2087:         MOVE.L  (SP)+,A0
2088:          BRA    J00403
2089: '
2090: J00292: MOVE.B  (A0)+,D0     ;．か NULL まで
2091:          BEQ.B  R293
2092:         CMP.B   #$2E,D0      ;'.'
2093:          BNE.B  J00292
2094:
2095:         MOVE.B  (A0),D0
2096:          BSR    CHK_10NUM    ;数字？
2097:          BCC.B  R293
2098:         SUBQ.L  #1,A0
2099: '数字でないので前に詰める
2100: P292_0:
2101:         MOVE.B  $0001(A0),(A0)+
2102:          BNE.B  P292_0
2103: R293:   RTS
2104:
2105:
2106: '1
2107: __IUSING:
2108:         MOVEM.L D0-D2/A1,-(SP)
2109:         LEA.L   -$0010(SP),SP
2110:         MOVE.L  A0,A1
2111:         MOVE.L  SP,A0
2112:          BSR.B  __LTOS
2113:         MOVE.L  SP,A0
2114:          BSR.B  _IUS_COUNT   ;文字数を数える
2115:         MOVEQ.L #$20,D2      ;
2116:         SUB.B   D0,D1        ;スペースの数
2117: P296:    BMI.B  J00297
2118:          BEQ.B  J00297
2119:         MOVE.B  D2,(A1)+     ;WRITE SPACE
2120:         SUBQ.B  #1,D1        ;COUNTER-1
2121:          BRA.B  P296
2122:
2123: J00297: MOVE.L  SP,A0
2124: P297_0:
2125:         MOVE.B  (A0)+,(A1)+  ;バッファーにデーター
を転送
2126:          BNE.B  P297_0
2127:         SUBQ.L  #1,A1
2128:         MOVE.L  A1,A0        ;バッファー位置更新
2129:         LEA.L   $0010(SP),SP
2130:         MOVEM.L (SP)+,D0-D2/A1
2131:         RTS
2132:
2133: _IUS_COUNT:
2134:         MOVEQ   #-1,D0
2135: P299:
2136:         ADDQ.W  #1,D0        ;+1
2137:         TST.B   (A0)+        ;CHECK
2138:          BNE.B  P299
2139:         SUBQ.L  #1,A0
2140:         RTS
2141: '1
2142: __LTOS:
2143:         MOVEM.L D0-D2/D7/A1-A2,-(SP)
2144:         LINK    A6,#-$000A
2145:         MOVE.L  SP,A1
2146:         LEA.L   L00313(PC),A2 ;変換用定数テーブル
2147:         CLR.W   D7           ;PLUS FLAG
2148:         TST.L   D0
2149:          BPL.B  P307
2150:         NEG.L   D0           ;絶体値化
2151:         NOT.W   D7           ;MINUS FLAG
2152: P307:
2153:         MOVE.L  (A2),D2      ;比較用定数
2154:          BEQ.B  J00309       ;終わり
2155:         CLR.B   D1
2156: P308:   ADDQ.B  #1,D1        ;変換中
2157:         SUB.L   D2,D0        ;定数と比較
2158:          BCC.B  P308         ;除算ＬＯＯＰ
2159:         ADD.L   D2,D0
2160:         SUBQ.B  #1,D1
2161:         MOVE.B  D1,(A1)+     ;WRITE DATA  答え
2162:         ADDQ.L  #4,A2
2163:          BRA.B  P307         ;ＬＯＯＰ
2164:
```

▶荻窪圭さんの「カウチポテト～」，略して「カポテチンデリデンブラブー」を読んで，僕は自分を見失った。記憶力がなくて，短気で，集中力のない僕はどーしたらいいんだー。
清水　恵三朗 (19) 東京都

```
2165: J00309: CLR.B   D1
2166:         MOVE.L  SP,A1
2167: P310:   TST.B   (A1)
2168:          BNE.B  J00311        ;終わりまで
2169:         ADDQ.L  #1,A1
2170:         ADDQ.B  #1,D1         ;文字数カウント
2171:         CMP.B   #$09,D1
2172:          BCS.B  P310
2173: J00311: TST.W   D7
2174:          BEQ.B  J00312
2175: 'マイナスの時
2176:         MOVE.B  #$2D,(A0)+    ;'-'
2177: J00312: MOVE.B  (A1)+,D0
2178:         ADD.B   #$30,D0       ;変換
2179:         MOVE.B  D0,(A0)+      ;数字 WRITE
2180:         ADDQ.B  #1,D1
2181:         CMP.B   #$0A,D1
2182:          BCS.B  J00312        ;10文字
2183:         CLR.B   (A0)          ;NULL
2184:         UNLK    A6
2185:         MOVEM.L (SP)+,D0-D2/D7/A1-A2
2186:         RTS
2187: '変換用データー
2188: L00313: .DC.L   $3B9ACA00     ;1000000000
2189:         .DC.L   $05F5E100     ;100000000
2190:         .DC.L   $0989680      ;10000000
2191:         .DC.L   $0F4240       ;1000000
2192:         .DC.L   $186A0        ;100000
2193:         .DC.L   $2710         ;10000
2194:         .DC.L   $3E8          ;1000
2195:         .DC.L   $64     ;100
2196:         .DC.L   $A      ;10
2197:         .DC.L   $1      ;1
2198:         .DC.L   $0
2199:
2200: *1
2201: __FVAL:
2202:         MOVE.L  D1,-(SP)      ;MOVEM.L
2203:         PEA      D_S_RET(PC)
2204: ***      BRA    S_DVAL
2205: __VAL:
2206: ¥¥***    BRA    S_DVAL
2207: S_DVAL:
2208:          BSR    SKIP_SPTAB
2209:         CMP.B   #$26,D0       ;'&'
2210:          BNE    S_STOD
2211:         ADDQ.L  #1,A0
2212:         MOVE.B  (A0)+,D0
2213:          BSR    TO_UPPER
2214:         CMP.B   #$42,D0       ;'B'
2215:          BEQ.B  J00315
2216:         CMP.B   #$4F,D0       ;'O'
2217:          BEQ.B  J00316
2218:         CMP.B   #$48,D0       ;'H'
2219:          BEQ.B  J00317
2220:         MOVE    #$0009,CCR    ;N=1 CY=1
2221:         RTS
2222: '2進
2223: J00315:
2224:          BSR    __STOB
2225:          BRA.B  J00318
2226: '8進
2227: J00316:
2228:          BSR    __STOO
2229:          BRA.B  J00318
2230: '16進
2231: J00317:
2232:          BSR    __STOH
2233: J00318:
2234:          BCS.B  R319
2235:          BSR    LONG_IEEE_DBL
2236:         MOVE    #$0000,CCR    ;0
2237: R319:   RTS
2238: *1
2239: __STOL:
2240:         MOVEM.L D1-D7,-(SP)
2241:         LINK    A6,#$0000
2242:         CLR.L   D0
2243:          BSR    SKIP_SPTAB
2244:          BEQ.B  J00323        ;変換出来ない
2245:          BSR    J00380
2246:         MOVE.B  (A0),D0
2247:          BSR    CHK_10NUM
2248:          BCS.B  J00323        ;変換出来ない
2249:         SUB.B   #$30,D0       ;数値
2250:         MCLRL   D1            ;
2251:         MOVE.B  D0,D1
2252: J00321: ADDQ.L  #1,A0
2253:         MOVE.B  (A0),D0       ;READ STRING
2254:          BSR    CHK_10NUM
2255:          BCS.B  J00324        ;終わり
2256:         SUB.B   #$30,D0       ;数値変換
2257:
2258:         MOVE.L  D1,-(SP)      ;*10
2259:         ADD.L   D1,D1         ;LSL.L  #1,D1
2260:          BCS.B  J00322
2261:         ADD.L   D1,D1         ;LSL.L  #1,D1
2262:          BCS.B  J00322
2263:         ADD.L   (SP)+,D1
2264:          BCS.B  J00322
2265:         ADD.L   D1,D1         ;LSL.L  #1,D1
2266:          BCS.B  J00322
2267:
2268:         ADD.L   D0,D1         ;+N
2269:          BCC.B  J00321
2270: 'オーバーフロー
2271: J00322: UNLK    A6
2272:         MOVEM.L (SP)+,D1-D7
2273:         MOVE    #$0003,CCR    ;CY=1 V=1
2274:         RTS
2275: '変換出来ない
2276: J00323: UNLK    A6
2277:         MOVEM.L (SP)+,D1-D7
2278:         MOVE    #$0009,CCR    ;CY=1 N=1
2279:         RTS
2280: '変換終了
2281: J00324: TST.W   D7
2282:          BMI.B  J00326
2283:         TST.L   D1
2284:          BMI.B  J00322        ;オーバーフロー
2285: J00325: MOVE.L  D1,D0
2286:         UNLK    A6
2287:         MOVEM.L (SP)+,D1-D7
2288:         MOVE    #$0000,CCR    ;0
2289:         RTS
2290: 'マイナスの時
2291: J00326: TST.L   D1
2292:          BPL.B  J00327
2293:         CMP.L   #$80000000,D1
2294:          BEQ.B  J00325
2295:          BRA.B  J00322        ;オーバーフロー
2296:
2297: J00327: NEG.L   D1            ;反転
2298:          BRA.B  J00325
2299: *2
2300: '8進変換
2301: __STOO:
2302:         MOVE.L  D1,-(SP)
2303:         MCLRL   D0            ;
2304:         MOVE.B  (A0),D0
2305:          BSR    CHK_8NUM
2306:          BCS.B  J00332        ;変換出来ない
2307:         SUB.B   #$30,D0       ;変換
2308:         MOVE.L  D0,D1
2309: P329:   ADDQ.L  #1,A0
2310:         MOVE.B  (A0),D0
2311:          BSR    CHK_8NUM
2312:          BCS.B  J00330        ;終わり
2313:
2314:         CMP.L   #$20000000,D1
2315:          BCC.B  J00331
2316:         LSL.L   #3,D1
2317:
2318:         SUB.B   #$30,D0
2319:         OR.B    D0,D1         ;+
2320:          BRA.B  P329
2321: '終わり
2322: J00330: MOVE.L  D1,D0
2323:         MOVE.L  (SP)+,D1
2324:         MOVE    #$0000,CCR    ;0
2325:         RTS
2326: 'オーバーフロー
2327: J00331: MOVE.L  (SP)+,D1
2328:         MOVE    #$0003,CCR    ;CY=1 V=1
2329:         RTS
2330: '変換出来ない
2331: J00332: MOVE.L  (SP)+,D1
2332:         MOVE    #$0009,CCR    ;CY=1 N=1
2333:         RTS
2334: *2
2335: '16進数変換
2336: __STOH:
2337:         MOVE.L  D1,-(SP)
2338:         MCLRL   D0            ;
2339:         MOVE.B  (A0),D0
2340:          BSR    CHK_HEX
2341:          BCS.B  J00332
2342:          BSR    TO_HEXVAL     ;16進変換
2343:         MOVE.L  D0,D1
2344: J00334: ADDQ.L  #1,A0
2345:         MOVE.B  (A0),D0
2346:          BSR    CHK_HEX
2347:          BCS.B  J00330
2348:         CMP.L   #$10000000,D1 ;*16
2349:          BCC.B  J00331
2350:         LSL.L   #4,D1
2351:          BSR    TO_HEXVAL
2352:         OR.B    D0,D1
2353:          BRA.B  J00334
2354: *2
2355: '2進変換
2356: __STOB:
2357:         MOVE.L  D1,-(SP)
2358:         MCLRL   D0            ;
2359:         MOVE.B  (A0),D0
2360:          BSR    CHK_2NUM
2361:          BCS.B  J00332        ;変換出来ない
2362:         SUB.B   #$30,D0
2363:         MOVE.L  D0,D1
2364: J00336: ADDQ.L  #1,A0
2365:         MOVE.B  (A0),D0
2366:          BSR    CHK_2NUM
2367:          BCS.B  J00330        ;終わり
2368:         ADD.L   D1,D1         ;LSL.L  #1,D1
2369:          BCS.B  J00331
2370:         SUB.B   #$30,D0
2371:         OR.B    D0,D1
2372:          BRA.B  J00336
2373: *1
2374: '2進文字へ変換
2375: __BTOS:
2376:         MOVEM.L D0-D2/A1,-(SP)
2377:         LINK    A6,#-$0022
2378:         MOVE.L  SP,A1
2379:         MOVEQ.L #$1F,D1       ;32 LOOP
2380: J00338: MOVEQ.L #$30,D2       ;'0'
2381:         ADD.L   D0,D0         ;LSL.L  #1,D0
2382:          BCC.B  P339
2383:         ADDQ.B  #1,D2
2384: P339:
2385:         MOVE.B  D2,(A1)+      ;WRITE TO BUFFER
2386:         DBF     D1,J00338
2387:          BRA.B  J00346
2388: *1
2389: '8進文字への変換
2390: __OTOS:
2391:         MOVEM.L D0-D2/A1,-(SP)
2392:         LINK    A6,#-$0022
2393:         MOVE.L  SP,A1
2394:         MOVEQ.L #$0A,D2       ;11 LOOP
2395:         ROL.L   #2,D0
2396:         MOVE.W  D0,D1
2397:         ANDI.W  #3,D1
2398:          BRA.B  J00342
2399:
2400: J00341:
2401:         ROL.L   #3,D0         ;3ビット取り出し
2402:         MOVE.W  D0,D1
2403: J00342:
2404:         ANDI.W  #7,D1
2405:
2406:         ADD.B   #$30,D1       ;文字
2407:         MOVE.B  D1,(A1)+      ;WRITE TO BUFF
2408:         DBF     D2,J00341
2409:          BRA.B  J00346
2410: *1
2411: '16進文字へ変換
2412: __HTOS:
2413:         MOVEM.L D0-D2/A1,-(SP)
2414:         LINK    A6,#-$0022
2415:         MOVE.L  SP,A1
2416:         MOVEQ.L #$07,D2
2417: J00344:
2418:         ROL.L   #4,D0         ;下位4ビット取り出し
2419:         MOVE    D0,D1
2420:         ANDI.W  #$0F,D1
2421:         CMP.B   #$0A,D1       ;9以上？
2422:          BCS.B  J00345
2423:         ADDQ.B  #7,D1         ;A-F 変換
2424: J00345: ADD.B   #$30,D1
2425:         MOVE.B  D1,(A1)+      ;WRITE TO BUFF
2426:         DBF     D2,J00344
2427: '変換終了
2428: J00346: CLR.B   (A1)          ;NULL追加
2429:         MOVE.L  SP,A1
2430: J00347: CMPI.B  #$30,(A1)     ;'0'
2431:          BNE.B  J00348
2432:         TST.B   $0001(A1)     ;NULL
2433:          BEQ.B  J00348
2434:         ADDQ.L  #1,A1         ;数値までポインターを
送る
2435:          BRA.B  J00347
2436:
2437: J00348: MOVE.B  (A1)+,(A0)+   ;コピーする
2438:          BNE.B  J00348
2439:         SUBQ.L  #1,A0
2440:         UNLK    A6
2441:         MOVEM.L (SP)+,D0-D2/A1
2442:         RTS
2443: *1
2444: __STOF:
2445:         MOVE.L  D1,-(SP)      ;MOVEM.L
2446:         PEA      D_S_RET(PC)
2447: ***      BRA    S_STOD
2448: __STOD:
2449: ¥¥***    BRA    S_STOD
2450: S_STOD:
2451:         MOVEM.L D2-D7/A1-A2,-(SP)
2452:         LINK    A6,#-$001E
2453:          BSR    SKIP_SPTAB
2454:          BEQ    J00357         ;END
2455:          BSR    J00380
2456:         MOVE.W  #$FFFF,-$0006(A6)   ;指数部バッファー
2457:         CLR.L   -$0004(A6)     ;仮数部バッファー
2458:         MOVE.B  (A0),D0
2459:         MCLRL   D6             ;
2460:         CMP.B   #$2E,D0        ;'.'
2461:          BNE.B  J00350
2462:
2463:          BSET   #31,D6  ;FLAG
2464:         CLR.W   -$0006(A6)
2465:         ADDQ.L  #1,A0
2466:         MOVE.B  (A0),D0
2467: J00350:  BSR    CHK_10NUM      ;数値？
2468:          BCS    J00357         ;終わり
2469:         ADDQ.L  #1,A0
2470:          BSR    J00360
2471:         LEA.L   -$001E(A6),A1
2472:         MOVE.B  #$30,(A1)+     ;'0'
2473:         MOVE.L  A1,A2
2474:         MOVE.B  D0,(A1)+
2475:         MOVEQ.L #$01,D5
2476: J00351: MOVE.B  (A0),D0
2477:         CMP.B   #$2E,D0        ;'.'
2478:          BNE.B  J00352
2479:         TST.L   D6
2480:          BNE    J00357         ;END
2481:          BSET   #31,D6         ;ピリオドFLAG
2482:         CLR.W   -$0006(A6)
2483:         ADDQ.L  #1,A0
2484:          BRA.B  J00351         ;LOOP
2485:
2486: J00352:  BSR    CHK_10NUM
2487:          BCS.B  J00353         ;数値でない
2488:         MOVE.B  D0,(A1)+       ;TO BUFF
2489:         ADDQ.L  #1,A0
2490:          BSR    J00360
2491:         ADDQ.W  #1,D5
2492:         CMP.W   #$000F,D5      ;15?
2493:          BNE.B  J00351         ;!= LOOP
2494:
2495:          BSR    J00374
2496: J00353:  BSR    J00367
2497:          BSR    J00362
2498:         TST.W   D7             ;符号
2499:          BPL.B  J00354
2500:          BSR    S_DNEG         ;符号反転
2501: J00354: MOVE.L  D0,-$000E(A6)
2502:         MOVE.L  D1,-$000A(A6)
2503:         TST.W   -$0006(A6)
2504:          BEQ.B  J00356
2505:          BSR    DBL_IEEE_LONG
2506:          BCS.B  J00355
2507:         MOVE.L  D0,D4
2508:          BSR    LONG_IEEE_DBL
2509:         MOVE.L  -$000E(A6),D2
2510:         MOVE.L  -$000A(A6),D3
2511:          BSR    IE_SDCMP
2512:          BNE.B  J00355
2513:         MOVE.L  D4,-$0004(A6)
2514:          BRA.B  J00356
2515:
2516: J00355: CLR.W   -$0006(A6)
2517: J00356: MOVE.L  -$000E(A6),D0
2518:         MOVE.L  -$000A(A6),D1
2519:         MCLRL   D2             ;
2520:         MOVE.W  -$0006(A6),D2
2521:         MOVE.L  -$0004(A6),D3
2522:         UNLK    A6
2523:         ADDQ.L  #8,SP
2524:         MOVEM.L (SP)+,D4-D7/A1-A2
2525:         MOVE    #$0000,CCR     ;0
2526:         RTS
2527:
2528: J00357: UNLK    A6
2529:         MOVEM.L (SP)+,D2-D7/A1-A2
2530:         MCLRL   D0             ;
2531:         MCLRL   D1             ;
2532:         MOVE    #$0009,CCR     ;CY=1 N=1
2533:         RTS
2534:
2535: J00358: UNLK    A6
2536:         MOVEM.L (SP)+,D2-D7/A1-A2
2537:         MOVE    #$0001,CCR     ;CY=1
2538:         RTS
2539:
2540: J00359: UNLK    A6
2541:         MOVEM.L (SP)+,D2-D7/A1-A2
2542:         MOVE    #$0003,CCR     ;CY=1 V=1
2543:         RTS
2544:
2545: J00360: TST.L   D6             ;フラグをみて指数部ー
1とする
2546:          BPL.B  R361
2547:         SUBQ.W  #1,D6
2548: R361:   RTS
2549:
2550: J00362: MCLRL   D0             ;
2551:         MCLRL   D1             ;
2552: J00363:
2553:         IE_REG  $20000000,$0402,D2,D3  ;*10 (=1.25*8)
2554:          BSR    IE_SDMUL
2555:         MOVE.L  D0,D2
2556:         MOVE.L  D1,D3
2557:         MCLRL   D0             ;
2558:         MOVE.B  (A2)+,D0       ;READ FROM BUFFER
2559:         SUB.B   #$30,D0
2560:          BSR    LONG_IEEE_DBL  ;数値を倍精度に変換
2561:          BSR    IE_SDADD
2562:         SUBQ.W  #1,D5
2563:          BNE.B  J00363         ;LOOP COUNT
2564:
2565:         ADD.W   D6,D4
2566:          BEQ.B  R366
2567:          BMI.B  J00365
2568: J00364:
2569:         IE_REG  $20000000,$0402,D2,D3  ;*10
2570:          BSR    IE_SDMUL
2571:          BCS.B  J00359         ;ERROR
2572:         SUBQ.W  #1,D4
2573:          BNE.B  J00364         ;LOOP
2574:         RTS
2575:
2576: J00365:
2577:         IE_REG  $20000000,$0402,D2,D3  ;÷10
2578:          BSR    IE_SDDIV
2579:          BCS.B  J00358         ;ERROR
2580:         ADDQ.W  #1,D4
2581:          BNE.B  J00365         ;LOOP
2582: R366:   RTS
2583:
2584: J00367: MCLRL   D4             ;
2585:         CMP.B   #$45,D0        ;'E'
2586:          BEQ.B  J00368
2587:         CMP.B   #$65,D0        ;'e'
2588:          BEQ.B  J00368
2589:         RTS
2590:
2591: J00368: CLR.W   -$0006(A6)
2592:         ADDQ.L  #1,A0
2593:         MOVE.B  (A0),D0
2594:         CMP.B   #$2B,D0        ;'+'
2595:          BEQ.B  J00369
2596:         CMP.B   #$2D,D0        ;'-'
2597:          BNE.B  J00370         ;PASS
```

▶2月号の特集を見て，音楽関係の記事が多いと勘違いしたのは，僕だけじゃないはず。しかし，このコピーを考えた人は，どこでレベッカとOh!Xの共通点をみつけたのだろうか？

大塩　芳正 (20) 愛知県

```
2598:          BSET   #31,D4          ;マイナスフラグ
2599: J00369: ADDQ.L #1,A0
2600:         MOVE.B (A0),D0
2601: J00370: BSR    CHK_10NUM
2602:          BCS   J00357           ;ERROR RETURN(0)
2603:         SUB.B  #$30,D0
2604:         MOVE.B D0,D4            ;数値
2605:         CLR.W  D0
2606: J00371: ADDQ.L #1,A0
2607:         MOVE.B (A0),D0
2608:          BSR   CHK_10NUM
2609:          BCS.B J00372           ;終わり
2610:         SUB.B  #$30,D0
2611:         MOVE.W D4,-(SP)
2612:         LSL.W  #2,D4
2613:         ADD.W  (SP)+,D4
2614:         ADD.W  D4,D4            ;LSL.W  #1,D4
2615:         ADD.W  D0,D4
2616:         CMP.W  #$03E8,D4
2617:          BCS.B J00371           ;LOOP
2618:         TST.L  D4               ;OVER
2619:          BPL   J00359           ;ERROR +
2620:          BRA   J00358           ;ERROR -
2621:
2622: J00372: TST.L  D4
2623:          BPL.B R373
2624:         NEG.W  D4
2625: R373:   RTS
2626:
2627: J00374: ADDQ.W #1,D6
2628:         SUBQ.W #1,D5
2629:         CMPI.B #$35,-(A1)       ;'5'
2630:          BCS.B J00376
2631: J00375: ADDQ.B #1,-(A1)
2632:         CMPI.B #$3A,(A1)        ;':'
2633:          BCS.B J00376
2634:         MOVE.B #$30,(A1)        ;'0'
2635:          BRA.B J00375           ;LOOP
2636:
2637: J00376: CMPA.L A2,A1
2638:          BCC.B P377
2639:         MOVE.L A1,A2
2640:         ADDQ.W #1,D6
2641: P377:   MOVE.B (A0),D0
2642:          BSR   CHK_10NUM        ;数値?
2643:          BCC.B J00378
2644:         CMP.B  #$2E,D0          ;'.'
2645:          BNE.B R379
2646:         TST.L  D6
2647:          BMI   J00357
2648:          BSET  #31,D6
2649: J00378: ADDQ.L #1,A0
2650:         TST.L  D6
2651:          BMI.B P377             ;LOOP
2652:         ADDQ.W #1,D6
2653:          BRA.B P377             ;LOOP
2654:
2655: R379:   RTS
2656:
2657: J00380: CLR.W  D7
2658:         CMP.B  #$2B,D0          ;'+'
2659:          BEQ.B J00381
2660:         CMP.B  #$2D,D0          ;'-'
2661:          BNE.B R382
2662:         NOT.W  D7
2663: J00381: ADDQ.L #1,A0
2664: R382:   RTS
2665:
2666: *SKIP
2667: J00383: ADDQ.L #1,A0
2668: SKIP_SPTAB:
2669:         MOVE.B (A0),D0
2670:         CMP.B  #$09,D0          ;HTAB
2671:          BEQ.B J00383
2672:         CMP.B  #$20,D0          ;' '
2673:          BEQ.B J00383           ;LOOP
2674:         TST.B  D0
2675:         RTS
2676:
2677: J00385: NEG.L  D0
2678:         ANDI.W #$00FF,D2
2679:         CMP.W  D2,D0
2680:          BLS.B P386
2681:         MOVE.W D2,D0
2682: P386:
2683:          BSR   R_MOV_BYTES
2684:         MOVE.B #$30,(A0)+       ;'0'
2685:          BSR   R_MOV_BYTES
2686:         MOVE.B #$2E,(A0)+       ;'.'
2687:         TST.W  D0
2688:          BEQ.B J00388
2689: J00387:
2690:          BSR   R_MOV_BYTES
2691:         MOVE.B #$30,(A0)        ;'0'
2692:         SUBQ.W #1,D0
2693:          BNE.B J00387           ;LOOP
2694: J00388:  BSR   J00403
2695:         MOVEM.L (SP)+,D1-D2
2696:         RTS
2697: *1
2698: _FGCVT:
2699:         MOVE.L D1,-(SP)         ;MOVEM.L
2700:          BSR   SNG_IEEE_DBL
2701:         PEA     D_S_RET0(PC)
2702: ***      BRA   S_GCVT
2703: *倍精度ー数値文字列
2704:  _GCVT:
2705: ******   BRA   S_GCVT          ;
2706: *倍精度を数値文字列に変換
2707: S_GCVT:
2708:         MOVEM.L D1-D2,-(SP)
2709:          BSR   S_ECVT
2710:         TST.L  D1
2711:          BEQ.B P390
2712:          BSR   R_MOV_BYTES
2713:         MOVE.B #$2D,(A0)+       ;'-'
2714: P390:   TST.L  D0
2715:          BEQ.B J00393
2716:          BMI.B J00391
2717:         ANDI.W #$00FF,D2
2718:         CMP.W  D2,D0
2719:          BEQ.B J00394
2720:          BCC.B J00392
2721:         LEA.L  $00(A0,D0.L),A0
2722:          BSR   R_MOV_BYTES
2723:         MOVE.B #$2E,(A0)        ;'.'
2724:          BSR   J00403
2725:          BSR   J00404
2726:          BRA.B J00394
2727:
2728: J00391: MOVE.L A0,-(SP)
2729:          BSR   J00403
2730:          BSR   J00404
2731:         SUBA.L (SP),A0
2732:         ADDQ.L #1,A0
2733:         MOVE.L D0,D1
2734:         NEG.L  D1
2735:         ADD.L  A0,D1
2736:         MOVE.L (SP)+,A0
2737:         ANDI.W #$00FF,D2
2738:         CMP.W  D2,D1
2739:          BCS   J00385
2740: J00392: ADDQ.L #1,A0
2741:          BSR.B R_MOV_BYTES
2742:         MOVE.B #$2E,(A0)        ;'.'
2743:          BSR.B J00403
2744:          BSR.B J00404
2745:         MOVE.B #$45,(A0)+       ;'E'
2746:         SUBQ.L #1,D0
2747:          BSR.B J00395
2748:         CLR.B  (A0)
2749:          BRA.B J00394
2750:
2751: J00393:  BSR.B R_MOV_BYTES
2752:         MOVE.B #$30,(A0)+       ;'0'
2753:          BSR.B R_MOV_BYTES
2754:         MOVE.B #$2E,(A0)+       ;'.'
2755:          BSR.B J00403
2756:          BSR.B J00404
2757: J00394:  BSR.B J00403
2758:         MOVEM.L (SP)+,D1-D2
2759:         RTS
2760:
2761: J00395: MOVE.B #$2B,D1          ;'+'
2762:         TST.L  D0
2763:          BPL.B J00396
2764:         MOVE.B #$2D,D1          ;'-'
2765:         NEG.L  D0
2766: J00396: MOVE.B D1,(A0)+
2767:         MOVE.W #$0064,D1        ;100
2768:          BSR.B J00397
2769:         MOVE.W #$000A,D1        ;10
2770:          BSR.B J00397
2771:         ADD.B  #$30,D0          ;+'0'
2772:         MOVE.B D0,(A0)+
2773:         RTS
2774: *商をバッファーに
2775: J00397: MOVE.B #$2F,D2          ;'0'-1
2776: P398:   ADDQ.B #1,D2
2777:         SUB.W  D1,D0
2778:          BCC.B P398             ;LOOP
2779:         ADD.W  D1,D0
2780:         MOVE.B D2,(A0)+
2781:         RTS
2782: *11
2783: *1BYTEづつ右移動　NULL終了
2784: R_MOV_BYTES:
2785:         MOVEM.L D0-D1/A0,-(SP)  ;NEW SPEED 1INST
2786:         MOVE.B (A0)+,D1
2787: R_MOV_B0:
2788:         MOVE.B D1,D0
2789:         MOVE.B (A0),D1          ;READ  N
2790:         MOVE.B D0,(A0)+         ;WRITE N-1
2791:          BNE.B R_MOV_B0
2792:         MOVEM.L (SP)+,D0-D1/A0
2793:         RTS
2794:
2795: J00402: ADDQ.L #1,A0
2796: J00403: TST.B  (A0)
2797:          BNE.B J00402           ;LOOP
2798:         RTS
2799:
2800: J00404: CMPI.B #$30,-(A0)       ;'0'
2801:          BEQ.B J00404
2802:         ADDQ.L #1,A0
2803:         CLR.B  (A0)
2804:         RTS
2805: *1
2806: _FECVT:
2807:          BSR   SNG_IEEE_DBL
2808: *****    BRA   S_ECVT  ;
2809: *1
2810: *倍精度ー数値文字列
2811:  _ECVT:
2812: ******   BRA   S_ECVT          ;
2813: *倍精度を数値文字列に変換
2814: S_ECVT: MOVEM.L D2-D7/A0-A2,-(SP)
2815:         LINK   A6,#-$0016
2816:          BSR   J00452
2817:         MOVE.W -$0012(A6),D0
2818:          BSR   J00454
2819:         MOVE.W -$0012(A6),D0
2820:          BSR   J00456
2821:         UNLK   A6
2822:         MOVE.W D7,D0
2823:         EXT.L  D0
2824:         MOVE.L D6,D1
2825:         MOVEM.L (SP)+,D2-D7/A0-A2
2826:         RTS
2827: *1
2828: _FFCVT:
2829:          BSR   SNG_IEEE_DBL
2830: *****    BRA   S_FCVT  ;
2831: *1
2832: *倍精度ー数値文字列
2833:  _FCVT:
2834: ***      BRA   S_FCVT          ;
2835: *倍精度を数値文字列に変換
2836: S_FCVT: MOVEM.L D2-D7/A0-A2,-(SP)
2837:         LINK   A6,#-$0016
2838:          BSR   J00452
2839:         MOVE.W -$0012(A6),D0
2840:         ADD.W  D7,D0
2841:          BSR   J00454
2842:         MOVE.W -$0012(A6),D0
2843:         ADD.W  D7,D0
2844:          BSR   J00456
2845:         UNLK   A6
2846:         MOVE.W D7,D0
2847:         EXT.L  D0
2848:         MOVE.L D6,D1
2849:         MOVEM.L (SP)+,D2-D7/A0-A2
2850:         RTS
2851: *1
2852: _FUSING:
2853:         MOVE.L D1,-(SP)         ;MOVEM.L
2854:          BSR   SNG_IEEE_DBL
2855:         PEA     D_S_RET(PC)
2856: ***      BRA   S_USING
2857:  _USING:
2858: *****    BRA   S_USING
2859: *2
2860: S_USING:
2861:         MOVEM.L D2-D7/A0-A2,-(SP)
2862:         TST.L  D3
2863:          BPL.B P408
2864:         CLR.L  D3
2865: P408:
2866:          BSR   IE_SDTST
2867:          BEQ.B P409
2868:          BTST  #3,D4
2869:          BNE.B J00413
2870: P409:
2871:          BTST  #4,D4
2872:          BEQ.B P410
2873:         ADDQ.L #1,D2
2874: P410:
2875:         MOVEM.L D2-D3,-(SP)
2876:         LINK   A6,#-$0016
2877:         MOVE.L D3,D2
2878:          BSR   J00452
2879:         MOVE.W -$0012(A6),D0
2880:         ADD.W  D7,D0
2881:          BSR   J00454
2882:         MOVE.W -$0012(A6),D0
2883:         ADD.W  D7,D0
2884:          BSR   J00456
2885:          BSR   J00427
2886:          BSR   J00430
2887:          BSR   J00432
2888:          BSR   J00435
2889:          BSR.B J00411
2890:          BSR   J00447
2891:          BSR   J00438
2892:          BSR   J00449
2893:         UNLK   A6
2894:         MOVE.W D7,D0
2895:         EXT.L  D0
2896:         MOVE.L D6,D1
2897:         ADDQ.L #8,SP
2898:         MOVEM.L (SP)+,D2-D7/A0-A2
2899:         RTS
2900:
2901: J00411:  BTST  #3,$0017(A6)
2902:          BEQ.B R412
2903:         MOVE.L -$0016(A6),A0
2904:          BSR   J00403
2905:         MOVE.B #$45,(A0)+       ;'E'
2906:         CLR.L  D0
2907:          BSR   J00395
2908:         CLR.B  (A0)
2909: R412:   RTS
2910:
2911: J00413:  BTST  #4,D4
2912:          BNE.B P414
2913:          BTST  #5,D4
2914:          BNE.B P414
2915:          BTST  #6,D4
2916:          BNE.B P414
2917:         TST.L  D2
2918:          BEQ.B P414
2919:         SUBQ.L #1,D2
2920:          BNE.B P414
2921:         TST.L  D3
2922:          BNE.B P414
2923:         MOVEQ.L #$01,D2
2924: P414:   MOVEM.L D2-D3,-(SP)
2925:         LINK   A6,#-$0016
2926:         ADD.L  D3,D2
2927:          BSR   J00452
2928:         MOVE.W -$0012(A6),D0
2929:          BSR   J00454
2930:         MOVE.W -$0012(A6),D0
2931:          BSR   J00456
2932:          BSR.B J00415
2933:         MOVE.L -$0016(A6),A0
2934:          BSR   J00403
2935:         MOVE.B #$45,(A0)+       ;'E'
2936:         MOVE.W D7,D0
2937:         EXT.L  D0
2938:          BSR   J00395
2939:         CLR.B  (A0)
2940:          BSR.B J00417
2941:         UNLK   A6
2942:         MOVE.W D7,D0
2943:         EXT.L  D0
2944:         MOVE.L D6,D1
2945:         ADDQ.L #8,SP
2946:         MOVEM.L (SP)+,D2-D7/A0-A2
2947:         RTS
2948:
2949: J00415: MOVE.L $0004(A6),D0
2950:         TST.L  $0010(A6)
2951:          BMI.B J00416
2952:         MOVE.L -$0016(A6),A0
2953:         LEA.L  $00(A0,D0.W),A0
2954:          BSR   R_MOV_BYTES
2955:         MOVE.B #$2E,(A0)        ;'.'
2956: J00416: SUB.W  D0,D7
2957:         RTS
2958:
2959: J00417: MOVE.L -$0016(A6),A0
2960:         MOVE.L $0014(A6),D0
2961:          BTST  #4,D0
2962:          BNE.B J00418
2963:          BTST  #5,D0
2964:          BNE.B J00423
2965:          BTST  #6,D0
2966:          BNE.B J00422
2967:         TST.L  D6
2968:          BNE.B J00420
2969:         MOVE.L $000C(A6),D1
2970:          BEQ.B R426
2971:         MOVEQ.L #$20,D0         ;' '
2972:         CMP.W  #$0002,D1
2973:          BCC.B J00421
2974:         TST.L  $0008(A6)
2975:          BEQ.B R426
2976:         MOVEQ.L #$30,D0
2977:          BRA.B J00421
2978:
2979: J00418: TST.L  D6
2980:          BNE.B J00420
2981:         MOVEQ.L #$2B,D0         ;'+'
2982:          BRA.B J00421
2983:
2984: J00419: MOVEQ.L #$20,D0         ;' '
2985:          BRA.B J00421
2986:
2987: J00420: MOVEQ.L #$2D,D0         ;'-'
2988: J00421:  BSR   R_MOV_BYTES
2989:         MOVE.B D0,(A0)
2990:         RTS
2991:
2992: J00422:  BSR   J00403
2993:         MOVEQ.L #$20,D0         ;' '
2994:          BRA.B J00424
2995:
2996: J00423:  BSR   J00403
2997:         MOVEQ.L #$2B,D0         ;'+'
2998: J00424: TST.L  D6
2999:          BEQ.B J00425
3000:         MOVEQ.L #$2D,D0         ;'-'
3001: J00425: MOVE.B D0,(A0)+
3002:         CLR.B  (A0)
3003: R426:   RTS
3004:
3005: J00427: TST.W  D7
3006:          BPL.B R429
3007:         NEG.W  D7
3008:         MOVE.L -$0016(A6),A0
3009:         CMP.W  -$0012(A6),D7
3010:          BCS.B J00428
3011:         MOVE.W -$0012(A6),D7
3012:          BEQ.B R429
3013: J00428:  BSR   R_MOV_BYTES
3014:         MOVE.B #$30,(A0)        ;'0'
3015:         SUBQ.W #1,D7
3016:          BNE.B J00428
3017: R429:   RTS
3018:
3019: J00430: TST.L  $0010(A6)
3020:          BMI.B R431
3021:         MOVE.L -$0016(A6),A0
3022:         LEA.L  $00(A0,D7.W),A0
3023:          BSR   R_MOV_BYTES
3024:         MOVE.B #$2E,(A0)        ;'.'
3025: R431:   RTS
3026:
3027: J00432: TST.W  D7
3028:          BNE.B R434
3029:         MOVE.L $000C(A6),D0
3030:          BEQ.B R434
3031:         SUBQ.L #2,D0
3032:          BCC.B J00433
```

▶エディタアセンブラ「REDA」。頭文字がRなので，てっきりリロケータブルアセンブラかと思いました。ZEDAを使っていたときは，ORGを忘れてよく暴走させたり，スペースを入れた数だけメモリが減るのを不満に思っていましたが，それらも改良されて使いやすそうですね。さっそく打ち込んでいます。

横田　紀明（21）山口県

```
3033:          MOVE.L  $0014(A6),D0
3034:          BTST    #4,D0
3035:          BNE.B   J00433
3036:          BTST    #5,D0
3037:          BNE.B   J00433
3038:          BTST    #6,D0
3039:          BNE.B   J00433
3040:          TST.L   D6
3041:          BNE.B   R434
3042: J00433:  MOVE.L  -$0016(A6),A0
3043:          BSR     R_MOV_BYTES
3044:          MOVE.B  #$30,(A0)        ;'0'
3045:          ADDQ.W  #1,D7
3046: R434:    RTS
3047:
3048: J00435:  BTST    #2,$0017(A6)
3049:          BEQ.B   R437
3050:          MOVE.L  -$0016(A6),A0
3051:          LEA.L   $00(A0,D7.W),A0
3052: J00436:  SUBQ.L  #3,A0
3053:          CMPA.L  -$0016(A6),A0
3054:          BLS.B   R437
3055:          BSR     R_MOV_BYTES
3056:          MOVE.B  #$2C,(A0)        ;','
3057:          ADDQ.W  #1,D7
3058:          BRA.B   J00436
3059:
3060: R437:    RTS
3061:
3062: J00438:  MOVE.L  -$0016(A6),A0
3063:          MOVE.L  $0014(A6),D0
3064:          BTST    #4,D0
3065:          BNE.B   J00442
3066:          BTST    #5,D0
3067:          BNE.B   J00443
3068:          BTST    #6,D0
3069:          BNE.B   J00444
3070:          TST.L   D6
3071:          BEQ.B   R441
3072: J00439:  MOVEQ.L #$2D,D0          ;'-'
3073: J00440:  BSR     R_MOV_BYTES
3074:          MOVE.B  D0,(A0)
3075:          ADDQ.W  #1,D7
3076: R441:    RTS
3077:
3078: J00442:  TST.L   D6
3079:          BNE.B   J00439
3080:          MOVEQ.L #$2B,D0          ;'+'
3081:          BRA.B   J00440
3082:
3083: J00443:  BSR     J00403
3084:          TST.L   D6
3085:          BNE.B   J00445
3086:          MOVEQ.L #$2B,D0          ;'+'
3087:          BRA.B   J00446
3088:
3089: J00444:  BSR     J00403
3090:          TST.L   D6
3091:          BNE.B   J00445
3092:          MOVEQ.L #$20,D0          ;' '
3093:          BRA.B   J00446
3094:
3095: J00445:  MOVEQ.L #$2D,D0          ;'-'
3096: J00446:  MOVE.B  D0,(A0)+
3097:          CLR.B   (A0)
3098:          RTS
3099:
3100: J00447:  BTST    #1,$0017(A6)
3101:          BEQ.B   R448
3102:          MOVE.L  -$0016(A6),A0
3103:          BSR     R_MOV_BYTES
3104:          MOVE.B  #$5C,(A0)        ;'¥'
3105:          ADDQ.W  #1,D7
3106: R448:    RTS
3107:
3108: J00449:  EXT.L   D7
3109:          SUB.L   $0004(A6),D7
3110:          BCC.B   R451
3111:          MOVE.L  -$0016(A6),A0
3112:          NEG.L   D7
3113:          MOVEQ.L #$20,D0          ;' '
3114:          BTST    #0,$0017(A6)
3115:          BEQ.B   J00450
3116:          MOVEQ.L #$2A,D0          ;'9'+1
3117: J00450:  BSR     R_MOV_BYTES
3118:          MOVE.B  D0,(A0)
3119:          SUBQ.L  #1,D7
3120:          BNE.B   J00450
3121: R451:    RTS
3122:
3123: J00452:  MOVE.L  A0,-$0016(A6)
3124:          CLR.W   -$0012(A6)
3125:          MOVE.B  D2,-$0011(A6)
3126:          BSR.B   J00461
3127:          LEA.L   -$0010(A6),A1
3128:          MOVE.B  #$30,(A1)+       ;'0'
3129:          MOVE.L  A1,-(SP)
3130:          BSR     J00467
3131:          LEA.L   -$0001(A6),A0
3132:          BSR     J00471
3133:          MOVE.L  (SP)+,A1
3134:          CMPA.L  A1,A0
3135:          BCC.B   R453
3136:          MOVE.L  A0,A1
3137:          ADDQ.W  #1,D7
3138: R453:    RTS
3139:
3140: J00454:  TST.W   D0
3141:          BMI.B   R455
3142:          CMP.W   #$000E,D0
3143:          BCC.B   R455
3144:          LEA.L   $00(A1,D0.W),A0
3145:          BSR     J00471
3146:          CMPA.L  A1,A0
3147:          BCC.B   R455
3148:          MOVE.L  A0,A1
3149:          ADDQ.W  #1,D7
3150: R455:    RTS
3151:
3152: J00456:  MOVE.L  -$0016(A6),A0
3153:          TST.W   D0
3154:          BEQ.B   J00460
3155:          BMI.B   J00460
3156:          CLR.W   D1
3157: J00457:  CMP.W   #$00FF,D1
3158:          BEQ.B   J00460
3159:          CMP.W   D0,D1
3160:          BEQ.B   J00460
3161:          CMP.W   #$000E,D1
3162:          BCC.B   J00459
3163:          MOVE.B  (A1)+,(A0)+
3164: J00458:  ADDQ.W  #1,D1
3165:          BRA.B   J00457
3166:
3167: J00459:  MOVE.B  #$30,(A0)+       ;'0'
3168:          BRA.B   J00458
3169:
3170: J00460:  CLR.B   (A0)
3171:          RTS
3172:
3173: J00461:  MCLRL   D6               ;
3174:          MOVE.L  D0,D2
3175:          ANDI.L  #$7FFFFFFF,D0    ;符号＋
3176:          BNE.B   J00462
3177:          TST.L   D1               ;D0.D1=0 ?
3178:          BNE.B   J00462
3179:          CLR.W   D7
3180:          RTS
3181:
3182: J00462:  ADD.L   D2,D2            ;LSL.L  #1,D2
3183:          ADDX.L  D6,D6            ;ROXL.L #1,D6
3184:          MOVEQ.L #$0F,D7
3185: J00463:
3186:          IE_REG  $635FA931,$A0000430,D2,D3
3187:          BSR     IE_SDCMP         ;比較
3188:          BCC.B   J00464
3189:          IE_REG  $00,$03FF,D2,D3 ; 1
3190:          BSR     IE_SDCMP         ;比較
3191:          BCC.B   J00465
3192:          IE_REG  $635FA931,$A0000430,D2,D3
3193:          BSR     IE_SDMUL         ;乗算
3194:          SUBI.W  #$000F,D7
3195:          BRA.B   J00463           ;LOOP
3196:
3197: J00464:
3198:          IE_REG  $635FA931,$A0000430,D2,D3
3199:          BSR     IE_SDDIV         ;除算
3200:          ADDI.W  #$000F,D7
3201:          BRA.B   J00463           ;LOOP
3202:
3203: J00465:
3204:          IE_REG  $35E620F4,$8000042D,D2,D3
3205:          BSR     IE_SDCMP         ;比較
3206:          BCC.B   R466
3207:          IE_REG  $20000000,$0402,D2,D3   ；１０
3208:          BSR     IE_SDMUL         ;乗算
3209:          SUBQ.W  #1,D7
3210:          BRA.B   J00465           ;LOOP
3211:
3212: R466:    RTS
3213:
3214: J00467:  MOVE.W  #$000E,D4
3215:          LEA.L   L00474(PC),A2
3216: J00468:  MOVE.B  #$30,(A1)        ;'0'
3217:          LEA.L   (A2),A0
3218: J00469:
3219:          MOVE.L  (A0),D2
3220:          MOVE.L  $0004(A0),D3
3221:          BSR     IE_SDCMP         ;比較
3222:          BCS.B   J00470
3223:          ADDQ.B  #1,(A1)
3224:          MOVE.L  (A0),D2
3225:          MOVE.L  $0004(A0),D3
3226:          BSR     IE_SDSUB         ;減算
3227:          BRA.B   J00469
3228:
3229: J00470:  ADDQ.L  #1,A1
3230:          ADDQ.L  #8,A2
3231:          DBF     D4,J00468
3232:          RTS
3233:
3234: J00471:  CMPI.B  #$35,(A0)        ;'5'  ４以下終わり
3235:          BCS.B   R473
3236: P472:
3237:          ADDQ.B  #1,-(A0)
3238:          CMPI.B  #$39,(A0)        ;'9'  ９以下終わり
3239:          BLS.B   R473
3240:          MOVE.B  #$30,(A0)        ;'0'  ０に換える
3241:          BRA.B   P472
3242:
3243: R473:    RTS
3244:
3245:          IE_DL   $635FA931,$A0000430        ;1000000000000000  17
桁
3246: L00474:  IE_DL   $35E620F4,$8000042D        ;100000000000000
3247:          IE_DL   $1184E72A,$0000042A        ;10000000000000
3248:          IE_DL   $68D4A510,$00000426        ;1000000000000
3249:          IE_DL   $3A43B740,$00000423        ;100000000000
3250:          IE_DL   $1502F900,$00000420        ;10000000000
3251:          IE_DL   $6E6B2800,$0000041C        ;1000000000
3252:          IE_DL   $3EBC2000,$00000419        ;100000000
3253:          IE_DL   $18968000,$00000416        ;10000000
3254:          IE_DL   $74240000,$00000412        ;1000000
3255:          IE_DL   $43500000,$0000040F        ;100000
3256:          IE_DL   $1C400000,$0000040C        ;10000
3257:          IE_DL   $7A000000,$00000408        ;1000
3258:          IE_DL   $48000000,$00000405        ;100
3259:          IE_DL   $20000000,$00000402        ;10
3260:          IE_DL   $00000000,$000003FF        ;1
3261:
3262: *1 IEEE
3263: __FPI:
3264:          MOVE.L  D1,-(SP)
3265:          PEA     D_S_RET(PC)
3266: ***      BRA     IE_SPI
3267: __PI:
3268: IE_SPI:
3269:          IE_REG  $490FDAA2,$2168C400,D0,D1       ;SHARP π
3270:          RTS
3271: *1 IEEE
3272: __FNPI:
3273:          MOVE.L  D1,-(SP)
3274:          BSR     SNG_IEEE_DBL
3275:          PEA     D_S_RET(PC)
3276: ***      BRA     IE_SNPI
3277: __NPI:
3278: ¥¥***    BRA     IE_SNPI
3279: IE_SNPI:
3280:          MOVEM.L D2-D3,-(SP)
3281:          IE_REG  $490FDAA2,$2168C400,D2,D3
3282:          BSR     IE_SDMUL                  ;乗算
3283: IE_POWRET:
3284:          MOVEM.L (SP)+,D2-D3
3285:          RTS
3286: *1
3287: __POWER:
3288:          MOVEM.L D2-D3,-(SP)
3289:          PEA     IE_POWRET(PC)
3290: ***      BRA.B   IE_SPOWER
3291: *2
3292: IE_SPOWER:
3293:          MOVEM.L D0-D1,-(SP)
3294:          MOVE.L  D2,D0
3295:          MOVE.L  D3,D1
3296:          BSR     IE_SDTST         ;テスト
3297:          BEQ.B   IE_483
3298:          BSR     IE_SDFRAC
3299:          BSR     IE_SDTST         ;テスト
3300:          BEQ.B   IE_486
3301:          MOVEM.L (SP)+,D0-D1
3302:          BSR     IE_SDTST         ;テスト
3303:          BEQ.B   IE_485
3304:          BSR     IE_SLOG
3305:          BCS.B   IE_R484
3306:          BSR     IE_SDMUL
3307:          BCS.B   IE_R484
3308:          BRA     IE_SEXP
3309:
3310: IE_483:  ADDQ.L  #8,SP
3311:          IE_REG  0,$3FF,D0,D1
3312: IE_R484:
3313:          RTS
3314:
3315: IE_485:  TST.L   D2
3316:          BPL.B   IE_R484
3317:          IE_REG  $7FFFFFFF,$FFFFFFFF,D0,D1
3318:          MOVE    #$0005,CCR       ;CY=1 Z=1
3319:          RTS
3320:
3321: IE_486:  MOVEM.L (SP)+,D0-D1
3322:          MOVEM.L D2-D3/D7,-(SP)
3323:          LINK    A6,#$0000
3324:          MOVEM.L D0-D1,-(SP)
3325:          MOVE.L  D2,-(SP)
3326:          BCLR    #31,D2
3327:          MOVE.L  D2,D0
3328:          MOVE.L  D3,D1
3329:          BSR     DBL_IEEE_LONG
3330:          BCS.B   IE_488
3331:          MOVE.L  D0,D7
3332:          BSR.B   IE_491
3333:          BCS.B   IE_488
3334:          TST.L   (SP)+
3335:          BPL.B   IE_487
3336:          MOVE.L  D0,D2
3337:          MOVE.L  D1,D3
3338:          IE_REG  0,$3FF,D0,D1
3339:          BSR     IE_SDDIV                  ;除算
3340: IE_487:  UNLK    A6
3341:          MOVEM.L (SP)+,D2-D3/D7
3342:          RTS
3343:
3344: IE_488:  MOVE    SR,D0
3345:          TST.L   -$000C(A6)
3346:          BPL.B   IE_489
3347:          BCHG    #1,D0
3348: IE_489:  BTST    #1,D0
3349:          BNE.B   IE_490
3350:          MCLRL   D0               ;
3351:          MCLRL   D1               ;
3352:          MOVE    #$0001,CCR       ;CY=1
3353:          BRA.B   IE_487
3354:
3355: IE_490:
3356:          IE_REG  $7FFFFFFF,$FFFFFFFF,D0,D1
3357:          MOVE    #$0003,CCR       ;CY=1 V=1
3358:          BRA.B   IE_487
3359:
3360: IE_491:  TST.L   D7               ;D7は整数
3361:          BEQ.B   IE_493
3362:          BTST    #0,D7
3363:          BEQ.B   IE_492
3364:          SUBQ.L  #1,D7
3365:          BEQ.B   IE_494
3366:          BSR.B   IE_491
3367:          BCS.B   IE_488
3368:          MOVE.L  -$0008(A6),D2
3369:          MOVE.L  -$0004(A6),D3
3370:          BRA     IE_SDMUL         ;乗算
3371:
3372: IE_492:  LSR.L   #1,D7
3373:          BSR.B   IE_491
3374:          BCS.B   IE_488
3375:          MOVE.L  D0,D2
3376:          MOVE.L  D1,D3
3377:          BRA     IE_SDMUL         ;乗算
3378:
3379: IE_493:  CLR.L   D0
3380:          CLR.L   D1
3381:          RTS
3382:
3383: IE_494:  MOVE.L  -$0008(A6),D0
3384:          MOVE.L  -$0004(A6),D1
3385:          RTS
3386: *1
3387: '単精度ＳＱＲ'平方根
3388: __FSQR:
3389:          MOVE.L  D1,-(SP)         ;MOVEM.L
3390:          BSR     SNG_IEEE_DBL
3391:          PEA     D_S_RET(PC)
3392: ****     BRA     IE_SSQR
3393: __SQR:
3394: ¥¥***    BRA     IE_SSQR
3395: IE_SSQR:
3396:          BSR     IE_SDTST
3397:          BEQ.B   IE_498           ;０の時
3398:          BMI.B   IE_499           ;負数はエラー
3399:          MOVEM.L D2-D3,-(SP)
3400:          MOVE.L  D0,D2
3401:          CLR.W   D2
3402:          SWAP    D2
3403:          MCLRL   D3               ;IEEE?危険
3404:          ANDI.L  #$FFFFF,D0       ;仮数部のみ
3405:          ANDI.W  #$7FF0,D2        ;指数部
3406:          SUBI.W  #$3FF0,D2
3407:          BTST    #4,D2
3408:          BEQ.B   IE_496
3409:          ADDI.W  #$10,D2          ;指数が奇数の時
3410:          ASR.W   #1,D2
3411:          ADDI.W  #$3FF0,D2
3412:          ORI.L   #$3FE00000,D0    ;仮数部埋める
3413:          BRA.B   IE_497
3414: IE_496:
3415:          ASR.W   #1,D2            ;指数が偶数の時
3416:          ADDI.W  #$3FF0,D2
3417:          ORI.L   #$3FF00000,D0    ;仮数部埋める
3418: IE_497:
3419: **¥      AND.W   #$7FF0,D2
3420:          SWAP    D2
3421:          BSR.B   IE_500
3422:          BSR     IE_SDMUL                  ;Ｘn＊Ｃ0 乗算
3423:          MOVEM.L (SP)+,D2-D3
3424:          RTS
3425: '０の時
3426: IE_498:  ANDI.B  #$FE,CCR         ;CY=0
3427:          RTS
3428: '負の時
3429: IE_499:  MCLRL   D0               ;０を返す
3430:          MCLRL   D1               ;
3431:          MOVE    #$0001,CCR       ;CY=1
3432:          RTS
3433:
3434: IE_500:  MOVEM.L D2-D5,-(SP)
3435:          MOVE.L  D0,D4            ;Ｘ0とする
3436:          MOVE.L  D1,D5
3437:          BSR     IE_SDADDONE      ;Ｘ0＋１
3438:          BSR     IE_SDDIVTWO      ;（Ｘ0＋１）÷２
3439: IE_501:  MOVEM.L D0-D1,-(SP)
3440:          MOVE.L  D0,D2
3441:          MOVE.L  D1,D3
3442:          MOVE.L  D4,D0
3443:          MOVE.L  D5,D1
3444:          BSR     IE_SDDIV         ;除算
3445:          MOVE.L  (SP),D2
3446:          MOVE.L  $0004(SP),D3
3447:          BSR     IE_SDADD         ;＋１
3448:          BSR     IE_SDDIVTWO      ;÷２
3449:          MOVEM.L (SP)+,D2-D3
3450:          BSR     IE_SDCMP         ;比較
3451:          BCS.B   IE_501           ;LOOP
3452:          MOVEM.L (SP)+,D2-D5
3453:          RTS
3454:
3455: *1
3456: '剰余 IEEE
3457: __CDMOD:
3458:          MOVEM.L D0-D3,-(SP)
3459:          MOVEM.L (A6),D0-D3
3460:          PEA     __C4RET(PC)
3461: *****    BRA     __DMOD
3462: __DMOD:
3463: ¥¥*      BRA     IE_SDMOD
3464: IE_SDMOD:
3465:          MOVEM.L D0-D3,-(SP)
3466:          BSR     IE_SDDIV                  ;除算
```

▶大学入試まであと18日。“でじたるざんまい”のおかげで２，３日は勉強できない。アセンブラのプログラムを組みたくなってしまったのだ。　東野　晃也（17）奈良県

```
3467:          BCS.B  IE_504
3468:          MOVE.W D0,D2
3469:          ANDI.L #$7FF00000,D2   ;商の指数部
3470:          CMP.L  #$43300000,D2
3471:          BCC.B  IE_503          ;大きすぎる
3472:          BSR    IE_SDFIX
3473:          MOVE.L $0008(SP),D2
3474:          MOVE.L $000C(SP),D3
3475:          BSR    IE_SDMUL              ;乗算
3476:          BCS.B  IE_504
3477:          MOVE.L D0,D2
3478:          MOVE.L D1,D3
3479:          MOVEM.L (SP)+,D0-D1
3480:          BSR    IE_SDSUB              ;減算
3481:          MOVEM.L (SP)+,D2-D3
3482:          RTS
3483:
3484: IE_503: MOVE   #$0003,CCR      ;CY=1 V=1
3485: IE_504: MOVEM.L (SP)+,D0-D3
3486:          RTS
3487: *1
3488: '倍精度符号
3489: _DSGN:
3490: *****    BRA    IE_SDSGN
3491: IE_SDSGN:
3492:          TST.L  D0
3493:          BPL.B  IE_DSGN_1
3494:          CMP.L  #$80000000,D0   ;マイナスの時
3495:          BNE.B  IE_DSGN_2
3496:          TST.L  D1
3497:          BEQ.B  IE_DSGN_0       ;マイナス0
3498: IE_DSGN_2:
3499:          MOVE.L #$BFF00000,D0   ;RETURN(-1)
3500:          MOVEQ  #0,D1
3501:          RTS
3502: IE_DSGN_0:
3503:          MCLRL  D0              ;RETURN(0) D0=D1=0
3504:          RTS
3505: IE_DSGN_1:
3506:          BNE.B  IE_P509         ;プラスの時
3507:          TST.L  D1
3508:          BEQ.B  IE_R510         ;
3509: IE_P509:
3510:          MOVEQ.L #$00,D1
3511:          MOVE.L #$3FF00000,D0   ;RETURN(1)
3512: IE_R510:
3513:          RTS
3514: *1
3515: '単精度符号
3516: _FSGN:
3517: *****    BRA    IE_SFSGN
3518: IE_SFSGN:
3519:          TST.L  D0
3520:          BEQ    IE_DSGN_0
3521:          BPL    IE_FSGN_1
3522:          CMP.L  #$80000000,D0
3523:          BEQ    IE_DSGN_0
3524:          MOVE.L #$BF800000,D0
3525:          RTS
3526: IE_FSGN_1:
3527:          MOVE.L #$3F800000,D0
3528:          RTS
3529:
3530: *1
3531: 'ＬＯＧ関数
3532: __FLOG:
3533:          MOVE.L D1,-(SP)        ;MOVEM.L
3534:          BSR    SNG_IEEE_DBL
3535:          PEA    D_S_RET(PC)
3536: ***      BRA    IE_SLOG
3537: _LOG:
3538: ****     BRA    IE_SLOG
3539: IE_SLOG:
3540:          BSR    IE_SDTST        ;テスト
3541:          BEQ    IE_515
3542:          BMI    IE_516
3543:          MOVEM.L D2-D7/A0-A2,-(SP)
3544:          LINK   A6,#$0000
3545:          MOVE.L D0,D4
3546:          MOVE.L D1,D5
3547:          SWAP   D0
3548:          LSR.W  #4,D0
3549:          ANDI.L #$07FF,D0       ;指数
3550:          SUBI.L #$03FF,D0
3551:          BSR    LONG_IEEE_DBL
3552:          IE_REG $317217F7,$D1CF7BFE,D2,D3
3553:          BSR    IE_SDMUL        ;乗算
3554:          BCS    IE_SER
3555:          EXG.L  D0,D4
3556:          EXG.L  D1,D5
3557:          SWAP   D0
3558:          AND.W  #$BFFF,D0
3559:          OR.W   #$3FF0,D0
3560:          SWAP   D0
3561:          BSR.B  IE_512
3562:          BCS    IE_SER
3563:          MOVE.L D4,D2
3564:          MOVE.L D5,D3
3565:          BSR    IE_SDADD        ;加算
3566:          BRA    IE_SER
3567:
3568: IE_512:
3569:          IE_REG $15F70000,$03FF,D2,D3
3570:          BSR    IE_SDCMP               ;比較
3571:          BCS    IE_EX_TBL4
3572:          IE_REG $54160000,$03FF,D2,D3
3573:          BSR    IE_SDCMP               ;比較
3574:          BCS.B  IE_513
3575:          SUBI.L #$100000,D0
3576:          BSR    IE_EX_TBL4
3577:          BCS.B  IE_R514
3578:          IE_REG $317217F7,$D1CF7BFE,D2,D3
3579:          BRA    IE_SDADD               ;加算
3580:
3581: IE_513:
3582:          IE_REG $3504F333,$F9DE63FE,D2,D3
3583:          BSR    IE_SDMUL               ;乗算
3584:          BCS.B  IE_R514
3585:          BSR    IE_EX_TBL4
3586:          BCS.B  IE_R514
3587:          IE_REG $317217F7,$D1CF7BFD,D2,D3
3588:          BRA    IE_SDADD               ;加算
3589: IE_R514:
3590:          RTS
3591:
3592: IE_515: MCLRL  D0              ;
3593:          MCLRL  D1              ;
3594:          MOVE   #$0005,CCR      ;CY=1 Z=1
3595:          RTS
3596:
3597: IE_516: MCLRL  D0              ;
3598:          MCLRL  D1              ;
3599:          MOVE   #$0001,CCR      ;CY-1
3600:          RTS
3601:
3602: IE_519:
3603:          TST.L  D4
3604:          BMI    IE_P537                ;0にしてリターン
3605: IE_518:
3606:          IE_REG $7FFFFFFF,$FFFFFFFF,D0,D1     ;最大値
3607:          MOVE   #$0003,CCR      ;CY=1 V=1
3608:          BRA    IE_SER
3609: *1
3610: '指数関数
3611: __FEXP:
3612:          MOVE.L D1,-(SP)        ;MOVEM.L
3613:          BSR    SNG_IEEE_DBL
3614:          PEA    D_S_RET(PC)
3615: ***      BRA    IE_SEXP
3616: _EXP:
3617: *****    BRA    IE_SEXP
3618: S_EXP:
3619: IE_SEXP:
3620:          MOVEM.L D2-D7/A0-A2,-(SP)
3621:          LINK   A6,#$0000
3622:          MOVE.L D0,D4
3623:          IE_REG $317217F7,$D1CF7BFE,D2,D3       ;0.6931......
3624:          BSR    IE_SDDIV                ;0.6931...で割る
3625:          BCS.B  IE_519
3626:
3627:          MOVE.L D0,D4                   ;整数部のみ取り出
す
3628:          MOVE.L D1,D5
3629:          BSR    IE_SDFIX
3630:
3631:          IE_REG $00,$0409,D2,D3         ;範囲をチェックす
る
3632:          BSR    IE_SDCMP                ;比較
3633:          BCC.B  IE_518
3634:          IE_REG $80000000,$0409,D2,D3
3635:          BSR    IE_SDCMP                ;比較
3636:          BLS    IE_P537
3637:
3638:          BSR    DBL_IEEE_LONG           ;整数に変換
3639:          MOVE.L D0,D6                   ;結果の指数部に加
算する用意
3640:
3641:          MOVE.L D4,D0                   ;少数部のみ取り出
し
3642:          MOVE.L D5,D1
3643:          BSR    IE_SDFRAC
3644:          IE_REG $317217F7,$D1CF7BFE,D2,D3       ;整数部のみに0
.6931..を掛ける
3645:          BSR    IE_SDMUL                ;乗算
3646:          BCS.B  IE_520
3647:
3648:          MOVE.L D0,D2                   ;指数部をチェック
する
3649:          SWAP   D2
3650:          ANDI.W #$7FF0,D2               ;指数部
3651:          SUBI.W #$10,D2
3652:          BCS.B  IE_520
3653:
3654:          SUBI.L #$100000,D0             ;指数－1
3655:          BSR    IE_EX_TBL3              ;級数展開
3656:          BCS.B  IE_520
3657:
3658:          MOVE.L D0,D2           ;二乗する
3659:          MOVE.L D1,D3
3660:          BSR    IE_SDMUL        ;乗算 Ｘ＊Ｘ
3661:          BCC.B  IE_521
3662: IE_520:
3663:          IE_REG $00,$03FF,D0,D1 ;＝１
3664: '―(      BRA    IE_521
3665: IE_521:
3666:          MOVE.L D0,D2           ;指数チェック後に合成
3667:          SWAP   D2
3668:          LSR.W  #4,D2
3669:          ANDI.L #$07FF,D2       ;指数部
3670:          ADD.W  D6,D2           ;指数部加算
3671:          BMI    IE_P537         ;小さすぎる0にする
3672:
3673:          CMP.W  #$07FF,D2
3674:          BCC    IE_518          ;大きすぎる
3675:
3676:          AND.L  #$800FFFFF,D0   ;仮数部
3677:          LSL.W  #4,D2
3678:          SWAP   D2
3679:          CLR.W  D2
3680:          OR.L   D2,D0           ;仮数部に指数部を合成
3681:          BRA    IE_SER
3682: *1
3683: '単精度ＡＴＡＮ関数
3684: __FATAN:
3685:          MOVE.L D1,-(SP)        ;MOVEM.L
3686:          BSR    SNG_IEEE_DBL
3687:          PEA    D_S_RET(PC)
3688: ***      BRA    IE_SATAN
3689: _ATAN:
3690: *****    BRA    IE_SATAN
3691: IE_SATAN:
3692:          MOVEM.L D2-D7/A0-A2,-(SP)
3693:          LINK   A6,#$0000
3694:          MOVE.L D0,-(SP)
3695:          BCLR   #31,D0
3696:          IE_REG $00,$03FF,D2,D3 ; 1
3697:          BSR    IE_SDCMP        ;比較
3698:          BCS.B  IE_523
3699:          EXG.L  D0,D2
3700:          EXG.L  D1,D3
3701:          BSR    IE_SDDIV        ;商
3702:          BSR.B  IE_525
3703:          BCS    IE_SER
3704:          MOVE.L D0,D2
3705:          MOVE.L D1,D3
3706:          IE_REG $490FDAA2,$2168C3FF,D0,D1
3707:          BSR    IE_SDSUB               ;減算
3708:          BRA.B  IE_524
3709:
3710: IE_523: BSR.B  IE_525
3711: IE_524: BCC    IE_535
3712:          BRA    IE_SER
3713:
3714: IE_525:
3715:          IE_REG $4BAD0000,$03FC,D2,D3
3716:          BSR    IE_SDCMP               ;比較
3717:          BCS    IE_EX_TBL2
3718:          IE_REG $2B090000,$03FE,D2,D3
3719:          BSR    IE_SDCMP               ;比較
3720:          BCC.B  IE_526
3721:          MOVE.L D0,D4
3722:          MOVE.L D1,D5
3723:          IE_REG $5413CCCF,$E77993FD,D2,D3
3724:          BSR    IE_SDMUL               ;乗算
3725:          BCS    IE_R527
3726:          BSR    IE_SDADDONE     ;＋１
3727:          BCS.B  IE_R527
3728:          EXG.L  D0,D4
3729:          EXG.L  D1,D5
3730:          IE_REG $5413CCCF,$E77993FD,D2,D3
3731:          BSR    IE_SDSUB               ;減算
3732:          BCS.B  IE_R527
3733:          MOVE.L D4,D2
3734:          MOVE.L D5,D3
3735:          BSR    IE_SDDIV               ;商
3736:          BCS.B  IE_R527
3737:          BSR    IE_EX_TBL2
3738:          BCS.B  IE_R527
3739:          IE_REG $450FDAA2,$2168C3FD,D2,D3
3740:          BRA    IE_SDADD               ;加算
3741:
3742: IE_526:
3743:          MOVE.L D0,D4
3744:          MOVE.L D1,D5
3745:          BSR    IE_SDADDONE     ;＋１
3746:          BCS.B  IE_R527
3747:          MOVE.L D4,D2
3748:          MOVE.L D5,D3
3749:          MOVE.L D0,D4
3750:          MOVE.L D1,D5
3751:          IE_REG $00,$03FF,D0,D1 ; 1
3752:          BSR    IE_SDSUB               ;減算
3753:          BCS.B  IE_R527
3754:          MOVE.L D4,D2
3755:          MOVE.L D5,D3
3756:          BSR    IE_SDDIV               ;商
3757:          BCS.B  IE_R527
3758:          BSR    IE_EX_TBL2
3759:          MOVE.L D0,D2
3760:          MOVE.L D1,D3
3761:          IE_REG $490FDAA2,$2168C3FE,D0,D1
3762:          BRA    IE_SDSUB        ;減算
3763: IE_R527:
3764:          RTS
3765: *1
3766: '単精度ＴＡＮ'ＴＡＮ関数
3767: __FTAN:
3768:          MOVE.L D1,-(SP)        ;MOVEM.L
3769:          BSR    SNG_IEEE_DBL
3770:          PEA    D_S_RET(PC)
3771: ***      BRA    IE_STAN
3772: _TAN:
3773: *****    BRA    IE_STAN
3774: IE_STAN:
3775:          MOVEM.L D2-D7/A0-A2,-(SP)
3776:          LINK   A6,#$0000
3777:          MOVE.L D0,D2
3778:          MOVE.L D1,D3
3779:          BSR.B  IE_SCOS         ;ＣＯＳ
3780:          BCS    IE_SER
3781:          EXG.L  D0,D2
3782:          EXG.L  D1,D3
3783:          BSR.B  IE_SSIN         ;ＳＩＮ
3784:          BCS    IE_SER
3785:          BSR    IE_SDDIV        ;除算 ＳＩＮ／ＣＯＳ
3786:          BRA    IE_SER
3787: *1
3788: '単精度ＣＯＳ'ＣＯＳ関数
3789: __FCOS:
3790:          MOVE.L D1,-(SP)        ;MOVEM.L
3791:          BSR    SNG_IEEE_DBL
3792:          PEA    D_S_RET(PC)
3793: ***      BRA    IE_SCOS
3794: _COS:
3795: ****     BRA    IE_SCOS
3796: IE_SCOS:
3797:          MOVEM.L D2-D7/A0-A2,-(SP)
3798:          LINK   A6,#$0000
3799:          BSR    IE_536
3800:          IE_REG $490FDAA2,$2168C3FF,D2,D3      ;π／２
3801:          BSR    IE_SDSUB        ;減算
3802:          BCS    IE_SER
3803:          BSR    IE_SDNEG        ;符号反転
3804:          BRA.B  IE_P531
3805: *1
3806: '単精度ＳＩＮ'ＳＩＮ関数
3807: __FSIN:
3808:          MOVE.L D1,-(SP)        ;MOVEM.L
3809:          BSR    SNG_IEEE_DBL
3810:          PEA    D_S_RET(PC)
3811: *****    BRA    IE_SSIN
3812: _SIN:
3813: *****    BRA    IE_SSIN
3814: IE_SSIN:
3815:          MOVEM.L D2-D7/A0-A2,-(SP)
3816:          LINK   A6,#$0000
3817:          BSR.B  IE_536
3818: IE_P531:
3819:          MOVE.L D0,-(SP)
3820:          BCLR   #31,D0          ;符号＋
3821:          IE_REG $490FDAA2,$2168C401,D2,D3      ;２π
3822:          BSR    IE_SDCMP        ;比較
3823:          BCS.B  IE_P532
3824:          BSR    IE_SDMOD        ;剰余
3825:          BCS.B  IE_SER
3826: IE_P532:
3827:          IE_REG $490FDAA2,$2168C400,D2,D3      ;π
3828:          BSR    IE_SDCMP        ;比較
3829:          BCS.B  IE_P533
3830:          BSR    IE_SDSUB        ;減算
3831:          BCS.B  IE_SER
3832:          BCHG   #7,(SP)
3833: IE_P533:
3834:          IE_REG $490FDAA2,$2168C3FF,D2,D3      ;π／２
3835:          BSR    IE_SDCMP        ;比較
3836:          BCS.B  IE_P534
3837:          IE_REG $450FDAA2,$2168C400,D2,D3      ;π
3838:          BSR    IE_SDSUB        ;減算
3839:          BCS.B  IE_SER
3840:          BCLR   #31,D0          ;符号＋
3841: IE_P534:
3842:          BSR.B  IE_540
3843:          BCS.B  IE_SER
3844: IE_535: TST.L  (SP)+
3845:          BPL.B  IE_SER
3846:          BSET   #31,D0          ;符号－
3847:          BRA.B  IE_SER
3848:
3849: IE_536:
3850:          TST.L  D0
3851:          BMI.B  IE_TS536
3852:          IE_REG $3EBC2000,$0419,D2,D3   ;＋
3853:          BSR    IE_SDCMP                ;最大値比較
3854:          BCC.B  IE_P537
3855:          RTS
3856: IE_TS536:
3857:          IE_REG $BEBC2000,$0419,D2,D3   ;－
3858:          BSR    IE_SDCMP                ;最小値比較
3859:          BHI.B  IE_R539
3860: IE_P537:
3861:          MOVEQ  #0,D0
3862:          MOVEQ  #0,D1
3863: IE_SER:
3864:          UNLK   A6
3865:          MOVEM.L (SP)+,D2-D7/A0-A2
3866: IE_R539:
3867:          RTS
3868:
3869: IE_540:
3870:          IE_REG $490FDAA2,$2168C3FE,D2,D3      ;π／４
3871:          BSR    IE_SDCMP        ;比較
3872:          BCS.B  IE_EX_TBL0
3873:          IE_REG $450FDAA2,$2168C3FF,D2,D3      ;π／２
3874:          BSR    IE_SDSUB        ;減算
3875:          BCLR   #31,D0          ;符号＋
3876:          BRA.B  IE_EX_TBL1
3877:
3878: IE_EX_TBL2:
3879:          LEA.L  S_TBL2(PC),A0
3880:          LEA.L  S_TBL3(PC),A1
3881:          BRA.B  IE_543
3882:
3883: IE_EX_TBL0:
3884:          LEA.L  S_TBL0(PC),A0
3885:          LEA.L  S_TBL1(PC),A1
3886: IE_543:
3887:          MOVE.L D0,D4
3888:          MOVE.L D1,D5
3889:          BSR.B  IE_545
3890:          BCC.B  IE_R544
3891:          MOVE.L D4,D0
3892:          MOVE.L D5,D1
3893: IE_R544:
3894:          RTS
3895:
```

▶先日，信州へスキーに行きました。今年の冬は寒いという長期予報はいったいどうなったんだ。1月だというのに，春のような暖かさで，おかげでチェーンはいらずにすんだけど，スキー板を傷だらけにしただけでした。気象庁のウソつき。吉岡　郁洋（30）奈良県

```
3896: IE_545:
3897:         MOVEM.L D0-D1,-(SP)     ;Xセーブ
3898:          BSR.B  IE_547
3899:          BCS.B  IE_P552
3900:         MOVEM.L (SP)+,D2-D3
3901:          BRA    IE_SDMUL        ;乗算
3902:
3903: IE_EX_TBL1:
3904:         LEA.L   S_TBL1(PC),A0
3905:         LEA.L   S_TBL2(PC),A1
3906: IE_547:
3907:         MOVE.L  D0,D2
3908:         MOVE.L  D1,D3
3909:          BSR    IE_SDMUL        ;二乗  X*X
3910:          BCC.B  IE_550
3911:         RTS
3912:
3913: IE_EX_TBL3:
3914:         LEA.L   S_TBL3(PC),A0
3915:         LEA.L   S_TBL4(PC),A1
3916:          BRA.B  IE_550
3917:
3918: IE_EX_TBL4:
3919:          BSR    IE_SDSUBONE     ;－1
3920:         LEA.L   S_TBL4(PC),A0
3921:         LEA.L   S_TBL5(PC),A1
3922: IE_550:
3923:         MOVEM.L D0-D1,-(SP)
3924:         MOVEM.L (A0)+,D0-D1     ;テーブル第1項
3925: IE_551:
3926:         MOVEM.L (SP),D2-D3
3927:          BSR    IE_SDMUL        ;乗算 N*X0
3928:          BCS.B  IE_P552
3929:
3930:         MOVEM.L (A0)+,D2-D3
3931:          BSR    IE_SDADD        ;加算 +(TBL)
3932:          BCS.B  IE_P552
3933:
3934:         CMPA.L  A1,A0
3935:          BNE.B  IE_551          ;LOOP
3936: IE_P552:
3937:         ADDQ.L  #8,SP
3938:         RTS
3939:
3940: S_TBL0:
3941:         IE_DL   $D73F9F39,$9DC0FBD6
3942:         IE_DL   $3092309D,$43684BDE
3943:         IE_DL   $D7322B3F,$AA2723E5
3944:         IE_DL   $38EF1D2A,$B639A3EC
3945:         IE_DL   $D00D00D0,$0D00D3F2
3946:         IE_DL   $08888888,$88888BF8
3947:         IE_DL   $AAAAAAAA,$AAAAABFC
3948:         IE_DL   $00000000,$000003FF             ;1
3949:
3950: S_TBL1:
3951:         IE_DL   $C9CBA546,$03E4EBDA
3952:         IE_DL   $0F76C77F,$C6C4C3E2
3953:         IE_DL   $93F27DBB,$C4FAE3E9
3954:         IE_DL   $500D00D0,$0D00D3EF
3955:         IE_DL   $B60B60B6,$0B60BBF5
3956:         IE_DL   $2AAAAAAA,$AAAAABFA
3957:         IE_DL   $80000000,$000003FE
3958:         IE_DL   $00000000,$000003FF
3959:
3960: S_TBL2:
3961:         IE_DL   $D79435E5,$0D7943FA
3962:         IE_DL   $70F0F0F0,$F0F0F3FA
3963:         IE_DL   $88888888,$88888BFB
3964:         IE_DL   $1D89D89D,$89D8A3FB
3965:         IE_DL   $BA2E8BA2,$E8BA33FB
3966:         IE_DL   $638E38E3,$8E38E3FB
3967:         IE_DL   $92492492,$492493FC
3968:         IE_DL   $4CCCCCCC,$CCCCD3FC
3969:         IE_DL   $AAAAAAAA,$AAAAABFD
3970:         IE_DL   $00000000,$000003FF             ;1
3971:
3972: S_TBL3:
3973:         IE_DL   $57322B3F,$AA2723E5
3974:         IE_DL   $13F27DBB,$C4FAE3E9
3975:         IE_DL   $38EF1D2A,$B639A3EC
3976:         IE_DL   $500D00D0,$0D00D3EF
3977:         IE_DL   $500D00D0,$0D00D3F2
3978:         IE_DL   $360B60B6,$0B60BBF5
3979:         IE_DL   $08888888,$88888BF8
3980:         IE_DL   $2AAAAAAA,$AAAAABFA
3981:         IE_DL   $2AAAAAAA,$AAAAABFC
3982:         IE_DL   $00000000,$000003FE             ;.5
3983:         IE_DL   $00000000,$000003FF             ;1
3984:         IE_DL   $00000000,$000003FF             ;1
3985:
3986: S_TBL4:
3987:         IE_DL   $E38E38E3,$8E38E3FA
3988:         IE_DL   $70F0F0F0,$F0F0F3FA
3989:         IE_DL   $80000000,$000003FB
3990:         IE_DL   $08888888,$88888BFB
3991:         IE_DL   $92492492,$492493FB
3992:         IE_DL   $1D89D89D,$89D8A3FB
3993:         IE_DL   $AAAAAAAA,$AAAAABFB
3994:         IE_DL   $3A2E8BA2,$E8BA33FB
3995:         IE_DL   $CCCCCCCC,$CCCCD3FB
3996:         IE_DL   $638E38E3,$8E38E3FB
3997:         IE_DL   $80000000,$000003FC
3998:         IE_DL   $12492492,$492493FC
3999:         IE_DL   $AAAAAAAA,$AAAAABFC
4000:         IE_DL   $4CCCCCCC,$CCCCD3FC
4001:         IE_DL   $80000000,$000003FD
4002:         IE_DL   $2AAAAAAA,$AAAAABFD
4003:         IE_DL   $80000000,$000003FE             ;-0.5
4004:         IE_DL   $00000000,$000003FF             ;1
4005:         IE_DL   $00000000,$00000000             ;0
4006: S_TBL5:
4007: *2
4008: __RANDOMIZE:
4009:         ADDI.L  #$00008000,D0
4010: *1
4011: __SRAND:
4012:         CMP.L   #$00010000,D0
4013:          BCS.B  J00560
4014:         MOVEQ.L #$FF,D0
4015:         RTS
4016:
4017: J00560: MOVE.L  D1,-(SP)        ;MOVEM.L
4018:         ANDI.L  #$0000FFFF,D0
4019:         LSL.L   #2,D0
4020:         MOVEQ.L #$01,D1
4021:          BCLR   #2,D0
4022:          BEQ.B  J00561
4023:         MOVEQ.L #$03,D1
4024: J00561:
4025:         OR.L    D1,D0
4026:         MOVE.L  D0,RND_BUFF
4027:         MOVEM.L (SP)+,D1
4028:         CLR.L   D0
4029:         RTS
4030: *1
4031: __RAND:
4032:          BSR.B  GEN_RND
4033:         ANDI.L  #$00007FFF,D0
4034:         RTS
4035: *3
4036: *整数乱数作成
4037: _GEN_RND:
4038:         MOVEM.L D1/A0,-(SP)
4039:         LEA.L   RND_BUFF(PC),A0
4040:         MOVE.L  (A0),D0
4041:         MOVE.L  D0,-(SP)        ;SAVE Rn
4042:         MOVE.L  #$00000383,D1
4043:          BSR    __UMUL          ;Rn+1=Rn*383
4044:         ANDI.L  #$0003FFFF,D0
4045:         MOVE.L  D0,(A0)         ;次回使用
4046:         MOVE.L  (SP)+,D0
4047:         MCLRL   D1              ;
4048:         LSR.L   #1,D0
4049:         LSR.L   #1,D0
4050:         ADDX.W  D1,D0           ;Rn/4 +cy
4051:         MOVEM.L (SP)+,D1/A0
4052:         RTS
4053:
4054: RND_BUFF:
4055:         ORI.B   #$01,D0         ;初期値
4056:
4057: *1
4058: *符号無し剰余
4059: __CUMOD:
4060:         MOVEM.L D0-D1,-(SP)
4061:         MOVEM.L (A6),D0-D1
4062:         PEA     _CDRET(PC)
4063: *****   BRA     __UMOD
4064: __UMOD:
4065:         MOVEM.L D1-D3/D7,-(SP)
4066:         MOVE.L  D0,D7
4067:          BEQ.B  UMOD_9          ;被除数が0ならそのまま
4068:         TST.L   D1
4069:          BEQ.B  _UMOD_ER        ;除数0ならエラー
4070:          BSR    SB_IDIV
4071:         MOVE.L  D2,D0           ;剰余を返す
4072: UMOD_9:
4073:         MOVEM.L (SP)+,D1-D3/D7
4074:         RTS
4075: *0除算エラー
4076: _UMOD_ER:
4077:         CLR.L   D0              ;ans=0
4078:         MOVEM.L (SP)+,D1-D3/D7
4079:         ORI.B   #$01,CCR        ;CY=1
4080:         RTS
4081:
4082: *1
4083: __CLTOF:
4084:         MOVE.L  D0,-(SP)        ;MOVEM.L
4085:         MOVE.L  (A6),D0         ;MOVEM.L
4086:         PEA     _CFRET(PC)
4087: *****   BRA     __LTOF
4088: *整数－単精度
4089: __LTOF:
4090: *****   BRA     LONG_IEEE_SNG
4091: *整数-単精度変換 IEEE
4092: LONG_IEEE_SNG:
4093:         TST.L   D0
4094:          BEQ.B  IE_R578         ;0ならそのまま
4095:         MOVEM.L D0-D1,-(SP)
4096:         ANDI.L  #$80000000,(SP) ;FLAGのみ  取り出す
4097:         MOVE.W  #$4B00,D1
4098:         TST.L   D0              ;符号
4099:          BPL.B  IE_576
4100:         NEG.L   D0              ;絶対値
4101: IE_576:
4102:         CMP.L   #$800000,D0
4103:          BCC    IE_576H         ;23ビット以上
4104: IE_5760:
4105:         BTST    #23,D0
4106:          BNE    IE_577          ;b23で終わり  仮数位置合わせ
4107:         SUBI.W  #$80,D1         ;指数－1
4108:         ADD.L   D0,D0           ;LSL.L #1,D0  ;仮数*2
4109:          BRA.B  IE_5760         ;LOOP
4110: IE_576HL:
4111:         LSR.L   #1,D0           ;23ビット以上の時
4112:         ADDI.W  #$80,D1
4113: IE_576H:
4114:         CMP.L   #$1000000,D0
4115:          BCC    IE_576HL
4116: IE_577:
4117:         SWAP    D1
4118:         CLR.W   D1
4119:         BCLR    #23,D0          ;仮数1ビット省略
4120:         OR.L    D1,D0           ;指数部
4121:         OR.L    (SP)+,D0        ;FLAG合成
4122:         MOVEM.L (SP)+,D1
4123: IE_R578:
4124:         RTS
4125: *1
4126: __CFTOL:
4127:         MOVE.L  D0,-(SP)        ;MOVEM.L
4128:         MOVE.L  (A6),D0         ;MOVEM.L
4129:         PEA     _CFRET(PC)
4130: *****   BRA     __FTOL
4131: *単精度－整数
4132: __FTOL:
4133: ****    BRA     SNG_IEEE_LONG
4134: *単精度－整数－変換 IEEE
4135: SNG_IEEE_LONG:
4136:         MOVEM.L D1-D2,-(SP)
4137:         MOVE.L  D0,D1
4138:         SWAP    D1
4139:         LSR.W   #7,D1
4140:         SUB.B   #$7F,D1         ;指数部変換
4141:          BCS.B  IE_582          ;<0
4142:         CMP.B   #$1F,D0
4143:          BCC.B  IE_583          ;大きすぎる
4144:         ANDI.W  #$FF,D1         ;シフト数D1
4145:         MOVE.L  D0,D2           ;仮数部分
4146:          BSET   #23,D0          ;省略ビットを立てる
4147:         ANDI.L  #$FFFFFF,D0     ;元の指数部0
4148:         CMP     #23,D1
4149:          BCC    IE_580H
4150: *22ビット以下の時
4151: IE_580:
4152:         LSR.L   #1,D0
4153:         ADDQ    #1,D1
4154:         CMP     #23,D1
4155:          BCS    IE_580
4156: IE_580E:
4157:         TST.L   D2
4158:          BPL.B  IE_P5800
4159:         NEG.L   D0              ;絶対値
4160: IE_P5800:
4161:         MOVEM.L (SP)+,D1-D2
4162:         RTS
4163: *20ビット以上の時
4164: IE_580HL:
4165:         ADD.L   D0,D0
4166:         SUBQ.W  #1,D1
4167: IE_580H:
4168:         CMP.W   #23,D1
4169:          BNE    IE_580HL
4170:         BRA     IE_580E
4171:
4172: *指数部小さすぎる
4173: IE_582:
4174:         MOVEM.L (SP)+,D1-D2
4175:         MOVEQ   #0,D0           ;0を返す
4176:         RTS
4177: *指数部が大きすぎる
4178: IE_583:
4179: IE_584:
4180:         MOVEM.L (SP)+,D1-D2
4181:         MOVE.L  #$7FFFFFFF,D0   ;最大値を返す
4182:         MOVE    #$0003,CCR      ;CY=1 V=1
4183:         RTS
4184:
4185: *1
4186: __CFTOD:
4187:         MOVEM.L D0-D1,-(SP)
4188:         MOVEM.L (A6),D0-D1
4189:         PEA     _CDRET(PC)
4190:          BRA.B  __FTOD
4191: SNG_EX_DBL:
4192:         EXG.L   D0,D1
4193: SNG_2_DBL:
4194:         MOVE.L  D1,D4
4195:          BSR    SNG_IEEE_DBL
4196:         MOVE.L  D0,D2
4197:         MOVE.L  D1,D3
4198:         MOVE.L  D4,D0
4199: ****     BRA    SNG_IEEE_DBL
4200: *単精度－倍精度
4201: __FTOD:
4202: *******  BRA    SNG_IEEE_DBL
4203: *単精度－倍精度－変換  IEEE
4204: SNG_IEEE_DBL:
4205:         TST.L   D0              ;値チェック
4206:          BEQ.B  IE_P589         ;0に変換
4207:         CMP.L   #$80000000,D0
4208:          BEQ.B  IE_P589         ;0に変換
4209:         MOVEQ   #0,D1
4210:         ASR.L   #1,D0
4211:         ROXR.L  #1,D1
4212:         ASR.L   #1,D0
4213:         ROXR.L  #1,D1
4214:         ASR.L   #1,D0
4215:         ROXR.L  #1,D1
4216:         SWAP    D0
4217:         ANDI.W  #$8FFF,D0
4218:         ADDI.W  #$3800,D0       ;指数部変換
4219:         SWAP    D0
4220:         RTS
4221:
4222: *0に変換
4223: IE_P589:
4224:         CLR.L   D1
4225:         RTS
4226:
4227: *1
4228: __CDTOF:
4229:         MOVEM.L D0-D1,-(SP)
4230:         MOVEM.L (A6),D0-D1
4231:         PEA     _CDRET(PC)
4232: *****    BRA    __DTOF
4233: *倍精度－単精度
4234: __DTOF:
4235: *****    BRA    DBL_IEEE_SNG
4236: *倍精度－単精度－変換 IEEE
4237: DBL_IEEE_SNG:
4238:         MOVE    SR,-(SP)
4239:         MOVEM.L D2,-(SP)
4240:          BCS.B  IE_627          ;エラー演算チェック
4241: IE_625:
4242:         MOVE.L  D0,D2
4243:         SWAP    D2
4244:         ANDI.W  #$7FF0,D2       ;指数部取り出し
4245:          BEQ.B  IE_626          ;0
4246:         SUBI.W  #$3800,D2       ;値が小さすぎる  精度オーバー 指数部精度変換
4247:         CMP.W   #$4800,D2       ;値が大きすぎる  精度オーバー
4248:          BCC.B  IE_M628
4249:         LSL.W   #4,D2           ;指数
4250:         ADD.L   D1,D1
4251:         ADDX.L  D0,D0
4252:         ROXR.W  #1,D2           ;フラグ合成
4253:         SWAP    D2
4254:         CLR.W   D2
4255:
4256:         ADD.L   D1,D1
4257:         ADDX.L  D0,D0           ;仮数上部
4258:         ADD.L   D1,D1
4259:         ADDX.L  D0,D0
4260:         ANDI.L  #$7FFFFF,D0
4261:         OR.L    D2,D0           ;フラグと指数合成
4262:         MOVE.L  (SP)+,D2
4263:         MOVE    (SP)+,CCR       ;SR
4264:         RTS
4265: *指数部0の時
4266: IE_626:
4267:         MOVE.L  (SP)+,D2
4268:         MOVE    (SP)+,CCR       ;SR
4269:         CLR.L   D0              ;0を返す
4270:         RTS
4271: *エラー演算後
4272: IE_627:
4273:          BEQ.B  IE_630
4274:          BMI.B  IE_631
4275:          BVS.B  IE_628          ;指数オーバー
4276:          BSR    IE_SDTST        ;値テスト
4277:          BEQ.B  IE_629          ;指数オーバー
4278:          BRA.B  IE_625          ;リカバリー
4279: IE_M628:
4280:         TST.W   D2
4281:          BMI.B  IE_629
4282: *値が大きすぎて  指数オーバー
4283: IE_628:
4284:         MOVE.L  (SP)+,D2
4285:         MOVE    (SP)+,CCR       ;SR
4286:         MOVE.L  #$7FFFFFFF,D0   ;最大値を返す
4287:         MOVE    #$0003,CCR      ;CY=1 V=1
4288:         RTS
4289: *値が小さすぎて  指数オーバー
4290: IE_629:
4291:         MOVE.L  (SP)+,D2
4292:         MCLRL   D0              ;0を返す
4293:         ADDQ    #2,SP           ;MOVE  (SP)+,CCR       ;SR
4294:         MOVE    #$0001,CCR      ;CY=1
4295:         RTS
4296:
4297: IE_630:
4298:         MOVE.L  (SP)+,D2
4299:         ADDQ    #2,SP           ;MOVE  (SP)+,CCR       ;SR
4300:         MOVE.L  #$7FFFFFFF,D0   ;最大値
4301:         MOVE    #$0005,CCR      ;CY=1 Z=1
4302:         RTS
4303:
4304: IE_631:
4305:         MOVE.L  (SP)+,D2
4306:         ADDQ    #2,SP           ;MOVE  (SP)+,CCR       ;SR
4307:         MOVE    #$0009,CCR      ;CY=1 N-1
4308:         RTS
4309:
4310:         .DATA
4311:
4312:         .BSS
4313:
4314: COMN_BUF:       .DS.B   $000000
4315: FLOAT_END:
4316:         .END    START
```

第78部 Z80用浮動小数点演算パッケージSOROBAN

●実数型パッケージ登場

いよいよ実数演算の登場です。これと同様なパッケージとして何本かの投稿もありました。最初にきたものは“IEEE準拠の単精度5バイト”と称するものでした。そのほか,とても速くて大きいもの,超高速1/2精度から倍精度まで,ドライバの差し換えで対応するという雄大な構想に立ったもの,無限桁まで対応しそうなもの,となかなかユニークなものがあったのですが,今回のSOROBANはもっともオーソドックスな部類に入ります。

その分,扱いやすく,機能的にもまとまっています。これなら,実数型コンパイラを作るのも,それほど困難ではないでしょう。当然,実数型SLANGに期待したいところですが,当面制作予定はないそうですので,投稿の狙い目かもしれません。

SOROBAN自体の,当面の課題は高速化です。誰かSOROBAN＋を作ってみませんか。求められる条件は,4Kバイトより大きくしないこと,バグパッチエリアを使い尽くさないこと,互換性があること,機種に依存しないこと,などです。

●OHM-Z80とは？

SOROBANは未発表のオリジナルアセンブラOHM-Z80でコーディングされています。作者が「史上最強のZ80用アブソリュートアセンブラ」というだけあって,なかなか強力なものなのですが,編集室では「もはやZ80用アセンブラではないのではないか」という意見も出ています。というのも,高級言語的な記述はもちろん,マクロによってあたかも「Z80は直交性に優れたCPUである」というふりをしているからです。これは初心者にもわかりやすそうな半面,ヘタに初心者が使うとZ80がわからなくなるという危険性をも含んでいます。

反響によっては,誌上公開も考えています。ぜひ,SOROBANのソースを読んだ感想を聞かせてください。もちろん,SOROBAN自体に対する意見もお待ちしています。

全機種共通システム掲載記事

■85年6月号
序論　共通化の試み
第1部　S-OS“MACE”
第2部　Lisp-85インタプリタ
第3部　チェックサムプログラム
■85年7月号
第4部　マシン語プログラム開発入門
第5部　エディタアセンブラZEDA
第6部　デバッグツールZAID
■85年8月号
第7部　ゲーム開発パッケージBEMS
第8部　ソースジェネレータZING
■85年9月号
インタラプト　S-OS番外地
第9部　マシン語入力ツールMACINTO-S
第10部　Lisp-85入門(1)
■85年10月号
第11部　仮想マシンCAP-X85
連載　Lisp-85入門(2)
■85年11月号
連載　Lisp-85入門(3)
■85年12月号
第12部　Prolog-85発表
■86年1月号
第13部　リロケータブルのお話
第14部　FM音源サウンドエディタ
■86年2月号
第15部　S-OS“SWORD”
第16部　Prolog-85入門(1)
■86年3月号
第17部　magiFORTH発表
連載　Prolog-85入門(2)
■86年4月号
第18部　思考ゲームJEWEL
第19部　LIFE GAME
連載　基礎からのmagiFORTH
連載　Prolog-85入門(3)
■86年5月号
第20部　スクリーンエディタE-MATE
連載　実戦演習magiFORTH
■86年6月号
第21部　Z80TRACER
第22部　magiFORTH TRACER
第23部　ディスクダンプ&エディタ
第24部　“SWORD”2000 QD
連載　対話で学ぶ magiFORTH
特別付録　PC-8801版S-OS“SWORD”
■86年7月号
第25部　FM音源ミュージックシステム
付録　FM音源ボードの製作
連載　計算力アップのmagiFORTH
特別付録　SMC-777版S-OS“SWORD”
■86年8月号
第26部　対局五目並べ
第27部　MZ-2500版S-OS“SWORD”
■86年9月号
第28部　FuzzyBASIC発表
連載　明日に向かってmagiFORTH
■86年10月号
第29部　ちょっと便利な拡張プログラム
第30部　ディスクモニタDREAM
第31部　FuzzyBASIC料理法<1>
■86年11月号
第32部　パズルゲームHOTTAN
第33部　MAZE in MAZE
連載　FuzzyBASIC料理法<2>
■86年12月号
第34部　CASL & COMET
連載　FuzzyBASIC料理法<3>
■87年1月号
第35部　マシン語入力ツールMACINTO-C
連載　FuzzyBASIC料理法<4>
■87年2月号
第36部　アドベンチャーゲームMARMALADE
第37部　テキアベ作成ツールCONTEX
■87年3月号
第38部　魔法使いはアニメがお好き
第39部　アニメーションツールMAGE
付録　“SWORD”再掲載とMAGICの標準化
■87年4月号
第40部　INVADER GAME
第41部　TANGERINE
■87年5月号
第42部　S-OS“SWORD”変身セット
第43部　MZ-700用“SWORD”をQD対応に
■87年6月号
インタラプト　コンパイラ物語
第44部　FuzzyBASICコンパイラ
第45部　エディタアセンブラZEDA-3
■87年7月号
第46部　STORY MASTER
■87年8月号
第47部　パズルゲーム碁石拾い
第48部　漢字出力パッケージJACKWRITE
特別付録　FM-7/77版S-OS“SWORD”
■87年9月号
第49部　リロケータブル逆アセンブラInside-R
特別付録　PC-8001/8801版S-OS“SWORD”
■87年10月号
第50部　tiny CORE WARS
第51部　FuzzyBASICコンパイラの拡張
第52部　X1turbo版S-OS“SWORD”
■87年11月号
序論　神話のなかのマイクロコンピュータ
付録　S-OSの仲間たち
第53部　もうひとつのFuzzyBASIC入門
第54部　ファイルアロケータ&ローダ
インタラプト　S-OSこちら集中治療室
第55部　BACK GAMMON
■87年12月号
第56部　タートルグラフィックパッケージTURTLE
第57部　X1turbo版“SWORD”アフターケア
ラインプリントルーチン
特別付録　PASOPIA7版S-OS“SWORD”
■88年1月号
第58部　FuzzyBASICコンパイラ・奥村版
付録　石上版コンパイラ拡張部の修正
■88年2月号
第59部　シューティングゲームELFES
■88年3月号
第60部　構造型コンパイラ言語SLANG
■88年4月号
第61部　デバッギングツールTRADE
第62部　シミュレーションウォーゲームWALRUS
■88年5月号
第63部　シューティングゲームELFES II
第64部　地底最大の作戦
■88年6月号
第65部　構造化言語SLANG入門(1)
第66部　Lisp-85用NAMPAシミュレーション
■88年7月号
第67部　マルチウィンドウドライバMW-1
連載　構造化言語SLANG入門(2)
■88年8月号
第68部　マルチウィンドウエディタWINER
■88年9月号
第69部　超小型エディタTED-750
第70部　アフターケアWINERの拡張
■88年10月号
第71部　SLANG用ファイル入出力ライブラリ
第72部　シューティングゲームMANKAI
■88年11月号
第73部　シューティングゲームELFES IV
■88年12月号
第74部　ソースジェネレータSOURCERY
■89年1月号
第75部　パズルゲームLAST ONE
第76部　ブロックゲームFLICK
■89年2月号
第77部　高速エディタアセンブラREDA
特別付録　X1版S-OS“SWORD”<再掲載>

*以上のアプリケーションは,基本システムであるS-OS“MACE”またはS-OS“SWORD”がないと動作しませんのでご注意ください。

全機種共通S-OS“SWORD”不要

Z80用浮動小数点演算パッケージ SOROBAN

大貫 信昭 *Ohnuki Nobuaki*

ついに登場，Z80用浮動小数点演算パッケージ，名づけて“SOROBAN”です。機種，システムに依存しないので，S-OSだけでなく，いろいろなシステムから使用することができます。

マシン語で実数を

パーソナルコンピュータは「電子計算機」とはいっても，数値演算（特に実数演算）のための特別なハードウェアを持っているわけではありません。8ビットCPUで一度に扱えるデータの大きさは1バイトか2バイトに限られています。整数を扱う場合なら2バイトサイズでも結構，実用になるのですが，実数を表現するためには4バイトから8バイトの大きさのデータを処理しなければなりません。

こうなると四則演算だけでも結構面倒な処理が必要になってきますし，我々が普段BASICでやっているような処理まで考えると，SINやLOGといった数値関数も必要です。プログラムを作るたびに，いちいちこのような部分までコーディングしていたのでは大変な労力がかかりますし，データ形式がマチマチになると扱いにくくなるので，このようなものはパッケージ化するのがいちばんです。

形式は?

問題はどのような形式でパッケージ化するかということでしょう。データの範囲が限定されているならば，固定小数点という形式で処理を軽くすることも考えられます。しかし，応用範囲を広くしなければパッケージ化の意味がありませんので，ここでは一般的な浮動小数点形式を採用します。さらに本格的なことをやろうとするなら，倍精度くらいの演算精度がほしいところでしょう。

結局，今回の浮動小数点演算パッケージで使用した浮動小数点のフォーマットはX1やMZと同じ単精度5バイト，倍精度8バイトのいわゆるシャープフォーマットに準拠しています。将来的にハードウェアでの実行を考えるならIEEE準拠にするということも考えられます（たとえば強引に8087を接続するということもできないわけではないのですから）。

IEEE準拠なら単精度4バイトですから，若干処理も軽くなりますが，精度的にはどうしても見劣りします。4バイトならデータをレジスタに持つことで大幅な高速化も不可能ではないでしょう。しかし，そうなると倍精度ルーチンとはまったく別の処理となりますので，サイズ的にはずいぶん不利になります。

現在のS-OSにとってメモリのフリーエリアはなによりも大切ですから，このようなパッケージはできるだけ小さくまとめなければなりません。また，このIEEE形式は世界標準とはいえ，シャープ系のマシンのユーザーにはあまりお馴染みでないでしょうし，シャープフォーマットならデバッグの際にX1turboのBIOSと比べることができるなどのメリットがあったので結局シャープフォーマットに準拠ということになりました。

図1　浮動小数点のフォーマット

SOROBANの仕様

さて，前述のようにこのSOROBANは単精度5バイト，倍精度8バイトをサポートしており，有効桁数は単精度8桁，倍精度16桁で扱える数値の範囲は2.94×10^{-39}～$\pm1.7\times10^{38}$となっています。

このようなパッケージに必要とされる高速，高精度，高機能，コンパクトの4つの条件をすべて満たすのは困難ですが，このパッケージでは精度，機能，コンパクトさともに満足できるレベルだと思います。問題は速度ですが，こちらのほうはまだまだ改善の余地があります（プログラムサイズとの兼ね合いもありますが）。

このプログラムでは乗除算がボトルネックになっています。X1turboのBIOSと比較すると3倍もの時間がかかってしまいます。そんなに違うことをやっているはずはないのにどうしてこんなに差があるのでしょうか？　この部分を高速化すればほかの関数も速くなるので，自信のある方はぜひ手を加えてみてください。

また，このSOROBANはS-OSシステムに依存していませんので，Z80のRAM上であればそのまま使用することが可能です。CP/Mや各機種のマシン語モニタ上で動作するプログラムから呼び出すこともできますので，幅広く使ってください。

SOROBANの使い方

リスト1はリロケータブルバイナリ形式(Oh!MZ1987年11月号参照）で出力されています。マシン語入力ツールでこれを打ち込んだら，

```
#S SOROBAN.L01:8000:94B3:1FFA
```

でセーブしておいてください。

実際に使用するときには，リスト2のロ

ーダを使って邪魔にならないアドレスにリロケートします。サンプルプログラムを使う場合は便宜上B000Hにあらかじめ SOROBANを読み込んでおいてください。

さらに，万一バグが発生した場合，リロケータブルオブジェクトでは対処に非常な困難が予想されますが，今回のプログラムではそのあたりにも対策が講じてあります。ソースリストのいちばん最後にそのためのデータ群が並んでいます。

SOROBANはHLレジスタと DE レジスタにデータの先頭アドレスを入れて，処理ルーチンを呼び出すと HL レジスタの指すアドレスから格納されているデータが変更されるという仕様になっています。パッケージの頭にジャンプテーブルがありますので呼び出すときはそこをコールしてください。各ルーチンの仕様を表にまとめてあります。実際の呼び出し方についてはサンプルプログラムを参考にしてください。

なお，このプログラムはオリジナルのアセンブラで開発されました。SLANGを作ったときのアセンブラとはまた違うバージョンです。自称究極のＺ80アセンブラなのですが，運がよければ日の目を見ることがあるかもしれません。ソースリストは参考程度に見ておいてください。来月はSLANGでこのパッケージを使うためのライブラリを発表する予定です。

Profile
◇大貫さんは栃木県にお住まいの28歳，会社員です。マイコン歴は約７年，X1turboユーザーです。構造化コンパイラSLANGの作者としてお馴染みですね。

表１　SOROBANの仕様

パッケージ仕様
1）単精度は５バイト，倍精度は８バイト使用する
2）有効桁数は，単精度８桁，倍精度16桁
3）データ，データ１はHLレジスタの指すアドレスに格納されている浮動小数点形式の数値を示す
4）データ２はDEレジスタの指すアドレスに格納されている浮動小数点形式の数値を示す
5）サブルーチン名の後ろの［ ］の中はパッケージの先頭アドレスからの相対アドレス。たとえば，パッケージがB000Hにあった場合，＃ADDのアドレスはB020Hとなる。

＃PRCSN［0000］
［機能］精度
［備考］内容が５のとき単精度，８のとき倍精度となる（ワークエリア）

＃MOVE［0002］
［機能］データ１をデータ２に代入する　［HL］＝＝＞［DE］
［入力］HL：データ１のアドレス
　　　DE：データ２のアドレス
［出力］DE：データ２のアドレス（結果）
［保存］AF以外のすべてのレジスタ

＃SWAP［0005］
［機能］データ１とデータ２を交換する　［HL］＜＝＞［DE］
［入力］HL：データ１のアドレス
　　　DE：データ２のアドレス
［出力］HL：データ１のアドレス（結果）
　　　DE：データ２のアドレス（結果）
［保存］AF以外のすべてのレジスタ

＃CVDBL［0008］
［機能］単精度のデータを倍精度にする
［入力］HL：データのアドレス
［出力］HL：データのアドレス（結果）
［保存］AF以外のすべてのレジスタ
［備考］バッファは８バイト必要
［例］入力：C0 2B 54 A8 BC　(1.2345678E＋19)
　　　出力：C0 2B 54 A8 BC 00 00 00 (1.234567800393669E＋19)

＃CVSNG［000B］
［機能］倍精度のデータを単精度にする
［入力］HL：データのアドレス
［出力］HL：データのアドレス（結果）
［保存］AF以外のすべてのレジスタ
［例］入力：C0 2B 54 A9 8C EB 1E EC (1.234567890123456E＋19)
　　　出力：C0 2B 54 A9 8D　(1.2345679E＋19)

＃CVSTF［000E］
［機能］ASCII文字列を浮動小数点形式のデータに変換
［入力］HL：データを格納するバッファのアドレス（5 or 8バイト）
　　　DE：ASCII文字列の先頭アドレス
［出力］HL：データのアドレス（結果）
　　　DE：ASCII文字列の終了アドレス＋１
［保存］AF，DE以外のすべてのレジスタ
［備考］関係のないコードが出てきたところでデータ終了とみなす
［例］12.3456
　　　−3
　　　42.6555E5
　　　1.2E−12

＃CVUTF［0011］
［機能］16ビット長無符号整数を浮動小数点形式のデータに変換
［入力］HL：データを格納するバッファのアドレス（5 or 8バイト）
　　　DE：16ビット長無符号整数
［出力］HL：データのアドレス（結果）
［保存］AF以外のすべてのレジスタ

＃CVITF［0014］
［機能］16ビット長符号付き整数を浮動小数点形式のデータに変換
［入力］HL：データを格納するバッファのアドレス（5 or 8バイト）
　　　DE：16ビット長符号付き整数
［出力］HL：データのアドレス（結果）
［保存］AF以外のすべてのレジスタ

＃CVFTS［0017］
［機能］浮動小数点形式のデータをASCII文字列に変換
［入力］HL：データのアドレス
　　　DE：文字列を格納するバッファのアドレス（18 or 34バイト）
［出力］DE：ASCII文字列の先頭アドレス（結果）
［保存］AF以外のすべてのレジスタ
［備考］エンドマークとして00Hが付く
［例］276.345
　　　0
　　　−9.999E−18

＃CVFTU［001A］
［機能］浮動小数点形式のデータを16ビット長無符号整数に変換
［入力］HL：データのアドレス
［出力］DE：16ビット長無符号整数（結果）
　　　Z：正なら１，負なら０
［保存］AF，DE以外のすべてのレジスタ

＃CVFTI［001D］
［機能］浮動小数点形式のデータを16ビット長符号付き整数に変換
［入力］HL：データのアドレス
［出力］DE：16ビット長符号付き整数（結果）
［保存］AF，DE以外のすべてのレジスタ

＃ADD［0020］
［機能］加算　［HL］＝［HL］＋［DE］
［入力］HL：データ１のアドレス
　　　DE：データ２のアドレス
［出力］HL：データ１のアドレス（結果）
［保存］AF以外のすべてのレジスタ

＃SUB［0023］
［機能］減算　［HL］＝［HL］−［DE］

[入力] HL：データ1のアドレス
DE：データ2のアドレス
[出力] HL：データ1のアドレス（結果）
[保存] AF以外のすべてのレジスタ

#MUL [0026]
[機能] 乗算 [HL]=[HL]*[DE]
[入力] HL：データ1のアドレス
DE：データ2のアドレス
[出力] HL：データ1のアドレス（結果）
[保存] AF以外のすべてのレジスタ

#DIV [0029]
[機能] 除算 [HL]=[HL]/[DE]
[入力] HL：データ1のアドレス
DE：データ2のアドレス
[出力] HL：データ1のアドレス（結果）
[保存] AF以外のすべてのレジスタ

#IDIV [002C]
[機能] 整数除算 [HL]=[HL]¥[DE]
[入力] HL：データ1のアドレス
DE：データ2のアドレス
[出力] HL：データ1のアドレス（結果）
[保存] AF以外のすべてのレジスタ

#MOD [002F]
[機能] 剰余算 [HL]=[HL]MOD[DE]
[入力] HL：データ1のアドレス
DE：データ2のアドレス
[出力] HL：データ1のアドレス（結果）
[保存] AF以外のすべてのレジスタ

#CMP [0032]
[機能] 比較 [HL]-[DE]
[入力] HL：データ1のアドレス
DE：データ2のアドレス
[出力] [HL]>[DE]の場合 A=1 CY=0 Z=0
[HL]=[DE]の場合 A=0 CY=0 Z=1
[HL]<[DE]の場合 A=-1 CY=1 Z=0
[保存] AF以外のすべてのレジスタ

#NEG [0035]
[機能] 符号反転 [HL]=-[HL]
[入力] HL：データのアドレス
[出力] HL：データのアドレス（結果）
[保存] AF以外のすべてのレジスタ

#INT [0038]
[機能] データを越えない最大の整数にする [HL]=INT[HL]
[入力] HL：データのアドレス
[出力] HL：データのアドレス（結果）
[保存] AF以外のすべてのレジスタ

#FIX [003B]
[機能] 小数点以下を切り捨てる [HL]=FIX[HL]
[入力] HL：データのアドレス
[出力] HL：データのアドレス（結果）
[保存] AF以外のすべてのレジスタ

#FRAC [003E]
[機能] 小数部をとる [HL]=FRAC[HL]
[入力] HL：データのアドレス
[出力] HL：データのアドレス（結果）
[保存] AF以外のすべてのレジスタ

#CINT [0041]
[機能] 小数第1位を四捨五入して整数にする [HL]=CINT[HL]
[入力] HL：データのアドレス
[出力] HL：データのアドレス（結果）
[保存] AF以外のすべてのレジスタ

#SQR [0044]
[機能] 平方根 [HL]=SQR[HL]
[入力] HL：データのアドレス
[出力] HL：データのアドレス（結果）
[保存] AF以外のすべてのレジスタ

#SIN [0047]
[機能] 三角関数SIN [HL]=SIN[HL]
[入力] HL：データのアドレス
[出力] HL：データのアドレス（結果）
[備考] 引数の値はラジアン単位
[保存] AF以外のすべてのレジスタ

#COS [004A]
[機能] 三角関数COS [HL]=COS[HL]
[入力] HL：データのアドレス
[出力] HL：データのアドレス（結果）
[備考] 引数の値はラジアン単位
[保存] AF以外のすべてのレジスタ

#TAN [004D]
[機能] 三角関数TAN [HL]=TAN[HL]
[入力] HL：データのアドレス
[出力] HL：データのアドレス（結果）
[備考] 引数の値はラジアン単位
[保存] AF以外のすべてのレジスタ

#ATN [0050]
[機能] アークタンジェント（逆正接） [HL]=ATN[HL]
[入力] HL：データのアドレス
[出力] HL：データのアドレス（結果）
[保存] AF以外のすべてのレジスタ
[備考] 引数の値は$-\pi/2 \sim \pi/2$

#EXP [0053]
[機能] 指数関数 [HL]=EXP[HL]
[入力] HL：データのアドレス
[出力] HL：データのアドレス（結果）
[保存] AF以外のすべてのレジスタ
[備考] 引数の値は88以下

#LOG [0056]
[機能] 自然対数 [HL]=LOG[HL]
[入力] HL：データのアドレス
[出力] HL：データのアドレス（結果）
[保存] AF以外のすべてのレジスタ

#POW [0059]
[機能] 累乗 [HL]=[HL]^[DE]
[入力] HL：データ1のアドレス
DE：データ2のアドレス
[出力] HL：データ1のアドレス（結果）
[保存] AF以外のすべてのレジスタ
[備考] データ1が負の場合，データ2は-65535～65535の整数

#PAI [005C]
[機能] 円周率πのデータ倍にする [HL]=PAI[HL]
[入力] HL：データのアドレス
[出力] HL：データのアドレス（結果）
[保存] AF以外のすべてのレジスタ

#RAD [005F]
[機能] 度単位からラジアン単位に変換する [HL]=RAD[HL]
[入力] HL：データのアドレス
[出力] HL：データのアドレス（結果）
[保存] AF以外のすべてのレジスタ

#ABS [0062]
[機能] 絶対値 [HL]=ABS[HL]
[入力] HL：データのアドレス
[出力] HL：データのアドレス（結果）
[保存] AF以外のすべてのレジスタ

#SGN [0065]
[機能] 符号を返す [HL]=SGN[HL]
[入力] HL：データのアドレス
[出力] HL：データのアドレス（結果）
データ：正の場合 1
0の場合 0
負の場合 -1
[保存] AF以外のすべてのレジスタ

リスト1 SOROBAN本体

```
8000 B4 04 03 00 06 00 09 00 : CA
8008 0C 00 0F 00 12 00 15 00 : 42
8010 18 00 1B 00 1E 00 21 00 : 72
8018 24 00 27 00 2A 00 2D 00 : A2
8020 30 00 33 00 36 00 39 00 : D2
8028 3C 00 3F 00 42 00 45 00 : 02
8030 48 00 4B 00 4E 00 51 00 : 32
8038 54 00 57 00 5A 00 5D 00 : 62
8040 60 00 63 00 66 00 69 00 : 92
8048 6C 00 6F 00 C3 00 C7 00 : 65
8050 CB 00 D2 00 DD 00 E0 00 : 5A
8058 EA 00 FB 00 FF 00 03 01 : E8
8060 06 01 0D 01 18 01 1B 01 : 4A
8068 24 01 29 01 2C 01 30 01 : AD
8070 4A 01 51 01 5E 01 61 01 : 5E
8078 6A 01 6D 01 76 01 79 01 : CA
---------------------------------
SUM: 63 08 FB 04 9D 04 D0 05 F614

8080 7C 01 83 01 94 01 97 01 : 2E
8088 9A 01 A6 01 AA 01 AD 01 : 9B
8090 B0 01 B3 01 B6 01 B9 01 : D6
8098 C5 01 CC 01 CF 01 D4 01 : 38
80A0 D7 01 01 02 08 02 0B 02 : F2
80A8 10 02 13 02 3D 02 40 02 : A8
80B0 43 02 47 02 4C 02 4F 02 : 2D
80B8 52 02 7F 02 82 02 85 02 : E0
80C0 C5 02 D1 02 D5 02 DF 02 : 52
80C8 E2 02 EE 02 F1 02 F5 02 : BE
80D0 FE 02 07 03 0C 03 19 03 : 35
80D8 24 03 27 03 31 03 34 03 : BC
80E0 39 03 3D 03 43 03 48 03 : 0D
80E8 4C 03 52 03 55 03 58 03 : 57
80F0 5B 03 61 03 64 03 68 03 : 94
80F8 6B 03 6F 03 72 03 76 03 : CE
---------------------------------
SUM: 1B 20 CE 22 47 22 8F 22 1D21

8100 7A 03 7F 03 84 03 90 03 : 19
8108 95 03 99 03 9C 03 9F 03 : 75
8110 A2 03 A6 03 A9 03 AC 03 : A9
8118 AF 03 BC 03 BF 03 CA 03 : 00
8120 CF 03 D2 03 D5 03 D8 03 : 5A
8128 E1 03 EF 03 F2 03 F6 03 : C4
8130 F9 03 FD 03 00 04 04 04 : 08
8138 08 04 0D 04 12 04 1F 04 : 56
8140 24 04 28 04 2B 04 2E 04 : B5
8148 31 04 34 04 37 04 3A 04 : E6
8150 44 04 49 04 52 04 55 04 : 44
8158 61 04 6A 04 6D 04 70 04 : B8
8160 73 04 78 04 7C 04 85 04 : FC
8168 8A 04 9A 04 9D 04 A2 04 : 73
8170 A5 04 AA 04 BA 04 BF 04 : D8
8178 C3 04 D1 04 D5 04 E1 04 : 5A
---------------------------------
SUM: 70 39 E1 39 2A 3A 8A 3A 786F

8180 EE 04 06 05 09 05 0E 05 : 1E
8188 1C 05 1F 05 25 05 28 05 : 9C
8190 30 05 35 05 38 05 3C 05 : ED
8198 44 05 47 05 4C 05 4F 05 : 3A
81A0 52 05 55 05 58 05 5B 05 : 6E
81A8 5E 05 61 05 64 05 67 05 : 9E
81B0 6A 05 6F 05 78 05 7B 05 : E0
81B8 7E 05 81 05 84 05 87 05 : 1E
81C0 8A 05 8D 05 90 05 93 05 : 4E
81C8 97 05 9A 05 9D 05 A2 05 : 84
81D0 A5 05 AC 05 B2 05 B5 05 : CC
81D8 B9 05 BC 05 C3 05 C6 05 : 12
81E0 D6 05 D9 05 DC 05 DF 05 : 7E
81E8 E2 05 ED 05 FF 05 03 06 : E6
81F0 28 06 2B 06 2F 06 33 06 : CD
81F8 41 06 44 06 48 06 4F 06 : 34
---------------------------------
SUM: B6 51 0B 52 5E 52 99 53 23C2

8200 52 06 59 06 61 06 64 06 : 88
8208 71 06 7A 06 7D 06 93 06 : 13
8210 99 06 9C 06 A1 06 A4 06 : 92
8218 A8 06 AD 06 C3 06 C6 06 : F6
8220 CB 06 D1 06 D4 06 DC 06 : 64
8228 DF 06 12 07 1F 07 2A 07 : 55
8230 2F 07 33 07 39 07 52 07 : 09
8238 58 07 5C 07 60 07 6F 07 : 9F
8240 95 07 99 07 9C 07 9F 07 : 85
8248 AA 07 B4 07 BC 07 D9 07 : 0F
8250 DC 07 ED 07 FE 07 01 08 : E5
8258 06 08 0D 08 1A 08 1F 08 : 6C
8260 29 08 35 08 38 08 3B 08 : F1
8268 42 08 45 08 48 08 4D 08 : 3C
8270 51 08 66 08 69 08 6D 08 : AD
8278 72 08 75 08 79 08 84 08 : 04
---------------------------------
SUM: 84 6F 2A 70 A0 70 39 71 F2FF

8280 8D 08 9D 08 A4 08 A7 08 : 95
8288 AA 08 AD 08 B0 08 B3 08 : DA
8290 B6 08 B9 08 BF 08 C8 08 : 16
8298 EA 08 FA 08 15 09 1F 09 : 3A
82A0 22 09 25 09 2B 09 2E 09 : C4
82A8 31 09 37 09 3B 09 3E 09 : 05
82B0 42 09 46 09 0D 0A 10 0A : CB
82B8 13 0A 17 0A 1B 0A 2B 0A : 98
82C0 2E 0A 38 0A 3B 0A 3F 0A : 08
82C8 42 0A 4E 0A 51 0A 56 0A : 5F
82D0 5A 0A 5D 0A 62 0A 65 0A : A6
82D8 6B 0A 74 0A 77 0A 7D 0A : FB
82E0 80 0A 84 0A 88 0A 8B 0A : 3F
82E8 91 0A 9A 0A 9E 0A A1 0A : 92
82F0 AC 0A B1 0A B4 0A 33 0B : 6D
82F8 36 0B 3B 0B 42 0B 46 0B : 25
---------------------------------
SUM: A7 96 17 96 37 98 04 99 2F53

8300 4D 0B 50 0B 53 0B 57 0B : 73
8308 5B 0B 5E 0B 61 0B 64 0B : AA
8310 67 0B 6A 0B 6E 0B 74 0B : DF
8318 77 0B 7D 0B 80 0B 83 0B : 23
8320 86 0B 8A 0B 8D 0B 90 0B : 59
8328 95 0B 98 0B 9B 0B 9E 0B : 92
8330 A1 0B A5 0B A8 0B AB 0B : C5
8338 B1 0B BA 0B BE 0B C1 0B : 16
8340 CC 0B D1 0B D4 0B DA 0B : 77
8348 DD 0B 96 0C 99 0C 9D 0C : D8
8350 A1 0C AF 0C BB 0C CB 0C : 06
8358 DA 0C DD 0C E0 0C E4 0C : AB
8360 E8 0C F4 0C F9 0C 00 0D : 06
8368 09 0D 10 0D 13 0D 19 0D : 79
8370 1C 0D 20 0D 23 0D 26 0D : B9
8378 29 0D 2C 0D 30 0D 33 0D : EC
---------------------------------
SUM: 4D B9 59 BA 97 BA E4 BB AECC

8380 39 0D 42 0D 46 0D 49 0D : 3E
8388 54 0D 5A 0D 5D 0D FA 0D : 39
8390 FD 0D 02 0E 07 0E 0A 0E : 47
8398 0D 0E 11 0E 15 0E 18 0E : 83
83A0 1B 0E 1F 0E 23 0E 26 0E : BB
83A8 2C 0E 35 0E 39 0E 3C 0E : 0E
83B0 47 0E 4C 0E 4F 0E 54 0E : 6E
83B8 5D 0E 60 0E 63 0E 66 0E : BE
83C0 6B 0E F8 0E FC 0E 01 0F : 99
83C8 0A 0F 0D 0F 10 0F 13 0F : 76
83D0 1F 0F 21 0F 23 0F 25 0F : C4
83D8 27 0F 29 0F 2B 0F 2D 0F : E4
83E0 2F 0F 31 0F 33 0F 35 0F : 04
83E8 37 0F 39 0F 3B 0F 3D 0F : 24
83F0 3F 0F 41 0F 43 0F 45 0F : 44
83F8 47 0F 49 0F 4B 0F 4D 0F : 64
---------------------------------
SUM: 29 E4 F2 E5 23 E5 EB E6 C005

8400 4F 0F 51 0F 53 0F 55 0F : 84
8408 57 0F 59 0F 5B 0F 5D 0F : A4
8410 5F 0F 61 0F 63 0F 65 0F : C4
8418 67 0F 69 0F 6B 0F 6D 0F : E4
8420 6F 0F 71 0F 73 0F 75 0F : 04
8428 77 0F 79 0F 7B 0F 7D 0F : 24
8430 7F 0F 81 0F 83 0F 85 0F : 44
8438 87 0F 89 0F 8B 0F 8D 0F : 64
8440 8F 0F 91 0F 93 0F 95 0F : 84
8448 97 0F 99 0F 9B 0F 9D 0F : A4
8450 9F 0F A1 0F A3 0F A5 0F : C4
8458 A7 0F A9 0F AB 0F AD 0F : E4
8460 AF 0F B1 0F B3 0F B5 0F : 04
8468 B7 0F B9 0F BB 0F BD 0F : 24
8470 BF 0F C1 0F C3 0F C5 0F : 44
8478 C7 0F C9 0F CB 0F CD 0F : 64
---------------------------------
SUM: B0 F0 D0 F0 F0 F0 10 F0 829D

8480 CF 0F D1 0F D3 0F D5 0F : 84
8488 D7 0F D9 0F DB 0F DD 0F : A4
8490 DF 0F E1 0F E3 0F E5 0F : C4
8498 E7 0F E9 0F EB 0F ED 0F : E4
84A0 EF 0F F1 0F F3 0F F5 0F : 04
84A8 F7 0F F9 0F FB 0F 00 00 : 18
84B0 00 00 00 00 08 00 C3 C0 : 8B
84B8 02 C3 CD 02 C3 81 06 C3 : A1
84C0 8F 06 C3 1D 07 C3 BE 06 : 03
84C8 C3 D6 06 C3 D4 07 C3 EC : EC
84D0 06 C3 11 07 C3 EA 02 C3 : 53
84D8 DB 02 C3 5D 03 C3 EB 03 : B1
84E0 C3 E1 05 C3 96 04 C3 B6 : 7F
84E8 04 C3 E2 06 C3 70 06 C3 : AB
84F0 E4 05 C3 26 06 C3 3F 06 : E0
84F8 C3 2C 05 C3 32 0A C3 27 : DD
---------------------------------
SUM: F5 93 77 52 67 93 7B 2C 3B79

8500 0A C3 0B 0A C3 29 0B C3 : 9C
8508 15 0D C3 F6 0D C3 88 0C : 3F
8510 C3 05 0F C3 08 0F C3 E9 : 5D
8518 0E C3 EE 0E C3 68 00 C3 : BB
8520 6B 00 C3 6E 00 B7 CB 1E : 3C
8528 23 CB 1E 23 CB 1E 23 CB : 06
8530 1E 23 CB 1E 23 CB 1E 23 : 59
8538 CB 1E 23 CB 1E C9 B7 CB : 40
8540 16 2B CB 16 2B CB 16 2B : 59
8548 CB 16 2B CB 16 2B CB 16 : F9
8550 2B CB 16 2B CB 16 C9 2B : 0C
8558 2B 2B CB 16 2B CB 16 2B : 6E
8560 CB 16 2B CB 16 2B CB 16 : F9
8568 C9 01 07 00 09 54 5D 2B : B6
8570 ED B8 AF 12 C9 D5 CD E8 : B9
8578 00 E1 11 9F 02 ED 4B 00 : CB
---------------------------------
SUM: 1F 8B 63 E9 C8 E4 19 12 4327

8580 00 ED B0 AF 12 21 A0 02 : 21
8588 3E 00 CB 7E 28 02 3E 01 : F0
8590 32 9E 02 3A 9F 02 B7 28 : 8C
8598 02 CB FE C9 ED 4B 00 00 : CC
85A0 79 FE 08 20 05 11 CB 1E : 9E
85A8 18 03 11 18 07 3D 32 B6 : 70
85B0 00 ED 53 72 00 7B 32 8B : EA
85B8 00 11 95 02 ED B0 AF 12 : 06
85C0 21 96 02 3E 00 CB 7E 28 : 68
85C8 02 3E 01 32 94 02 3A 95 : D8
85D0 02 B7 28 02 CB FE C9 3A : AF
85D8 95 02 B7 C8 21 96 02 3A : 09
85E0 95 02 47 3A 00 00 4F 11 : 78
85E8 08 00 7E B7 20 0E 14 23 : A2
85F0 78 93 47 28 62 38 60 0D : 81
85F8 28 5D 18 EE 78 32 95 02 : CC
---------------------------------
SUM: FA D4 82 1D 39 C2 4E 10 59F5

8600 7A B7 28 0D 11 96 02 06 : 15
8608 00 ED B0 47 AF 12 13 10 : C8
8610 FB 11 96 02 3A 95 02 47 : BC
8618 1A CB 7F 20 0B 21 9D 02 : 4F
8620 CD 8A 00 05 28 31 18 F0 : BD
8628 78 32 95 02 3A 00 00 21 : 9C
8630 9A 02 FE 08 20 03 21 9D : 83
8638 02 CB 7E 28 0D 3D 47 37 : 3B
8640 2B 7E CE 00 77 10 F9 DC : D3
8648 3C 02 21 96 02 3A 94 02 : C7
8650 FE 00 20 02 CB BE C9 3E : B0
8658 00 32 94 02 AF 32 95 02 : 40
8660 21 03 0A 11 96 02 C3 C0 : 5A
8668 02 21 95 02 3A 00 00 36 : 2A
8670 FF 23 3D 20 FA 36 00 C9 : 78
8678 3A 00 00 FE 05 20 08 11 : 76
---------------------------------
SUM: 31 02 7D 78 56 61 EA 32 8E94

8680 9A 02 21 A4 02 18 15 11 : A1
8688 9D 02 21 A7 02 1A 96 12 : 2B
8690 1B 2B 1A 9E 12 1B 2B 1A : 70
8698 9E 12 1B 2B 1A 9E 12 1B : DB
86A0 2B 1A 9E 12 1B 2B 1A 9E : F3
86A8 12 1B 2B 1A 9E 12 1B 2B : 68
86B0 1A 9E 12 C9 3A 00 00 FE : CB
86B8 05 20 08 11 9A 02 21 A4 : 9F
86C0 02 18 15 11 9D 02 21 A7 : A7
86C8 02 1A 86 12 1B 2B 1A 8E : A2
86D0 12 1B 2B 1A 8E 12 1B 2B : 58
86D8 1A 8E 12 1B 2B 1A 8E 12 : BA
86E0 1B 2B 1A 8E 12 1B 2B 1A : 60
86E8 8E 12 1B 2B 1A 8E 12 D0 : 70
86F0 21 96 02 CD 72 00 21 95 : AE
86F8 02 34 CC B5 01 37 C9 11 : C9
---------------------------------
SUM: 48 16 35 AD CD 63 49 C5 653E

8700 96 02 21 A0 02 3A 00 00 : 95
8708 FE 08 20 0F 1A BE C0 13 : E0
8710 23 1A BE C0 13 23 1A BE : C9
8718 C0 13 23 1A BE C0 13 23 : C4
8720 1A BE C0 13 23 1A BE C0 : 66
8728 13 23 1A BE C0 13 23 1A : 1E
8730 BE C9 21 94 02 11 9E 02 : EF
8738 3A 00 00 47 04 04 4E 1A : F1
8740 77 79 12 23 13 10 F7 C9 : 08
8748 00 00 00 00 00 00 00 00 : 00
8750 00 00 00 00 00 00 00 00 : 00
8758 00 00 00 00 00 00 00 00 : 00
8760 00 00 00 00 00 00 00 00 : 00
8768 00 00 00 00 00 00 00 00 : 00
8770 00 00 00 00 C5 D5 E5 ED : 6C
8778 4B 00 00 ED B0 E1 D1 C1 : 5B
---------------------------------
SUM: 5E 5A 2F 45 5E E3 67 61 7B43

8780 C9 C5 D5 E5 3A 00 00 47 : C9
8788 CD 8A 02 E1 D1 C1 C9 C5 : 5A
8790 D5 E5 CD C1 00 21 9E 02 : 09
8798 7E EE 01 77 18 06 C5 D5 : 9C
87A0 E5 CD C1 00 3A 95 02 47 : 8B
87A8 3A 9F 02 4F 78 B9 30 08 : 93
87B0 C5 CD 7E 02 C1 78 41 4F : DB
87B8 0C 0D CA 72 04 16 28 3A : D1
87C0 00 00 FE 08 20 02 16 40 : 7E
87C8 78 91 47 BA D2 72 04 0E : 60
87D0 08 78 91 38 0B 47 C5 21 : 81
87D8 A0 02 CD B5 00 C1 18 F1 : EE
87E0 04 05 28 08 21 A0 02 CD : C9
87E8 71 00 10 F8 3A 94 02 47 : 90
87F0 3A 9E 02 B8 20 05 CD 00 : 84
87F8 02 18 13 3A 95 02 47 3A : 7F
---------------------------------
SUM: AA 2E A0 62 A7 7B D6 69 5694

8800 9F 02 B8 20 06 CD 4B 02 : 99
8808 DC 7E 02 CD C4 01 C3 72 : 23
8810 04 C5 D5 E5 CD C1 00 3A : 4B
8818 95 02 B7 CA 72 04 3A 9F : 67
8820 02 B7 CA 84 04 3A 9E 02 : E5
8828 4F 3A 94 02 A9 32 94 02 : 90
8830 26 00 3A 95 02 6F 54 3A : F4
8838 9F 02 5F 19 11 7F FF 19 : C1
8840 7C FE FF CA 84 04 FE 01 : CA
8848 CA 89 04 7D 32 95 02 21 : BE
8850 96 02 11 E3 03 CD C0 02 : 1E
8858 EB 21 03 0A CD C0 02 21 : C9
8860 E3 03 3A 00 00 3D 47 4E : F2
8868 23 C5 E5 0C 0D 20 08 21 : 2F
8870 A0 02 CD B5 00 18 19 06 : 5B
8878 08 CB 21 30 0B CD 00 02 : FE
---------------------------------
SUM: 9F 79 61 F5 67 55 F7 60 EA5E

8880 30 06 21 A0 02 CD 71 00 : 37
```

```
8888 21 A0 02 CD 71 00 10 E9 : FA
8890 E1 C1 10 D3 C3 72 04 00 : BE
8898 00 00 00 00 00 00 00 C5 : C5
88A0 D5 E5 CD C1 00 3A 95 02 : 19
88A8 B7 CA 72 04 3A 9F 02 B7 : 89
88B0 CA 84 04 3A 9E 02 4F 3A : B5
88B8 94 02 A9 32 94 02 26 00 : 2D
88C0 3A 95 02 6F 54 3A 9F 02 : 6F
88C8 5F B7 ED 52 1E 81 19 7C : 89
88D0 FE FF CA 84 04 FE 01 CA : 18
88D8 89 04 7D 32 95 02 21 96 : 8A
88E0 02 CD 71 00 21 A0 02 CD : D0
88E8 71 00 21 8E 04 3A 00 00 : 5E
88F0 3D 47 C5 E5 01 00 08 CD : 04
88F8 4B 02 38 06 CD C4 01 37 : 54
---------------------------------
SUM: 37 01 E4 61 A0 75 76 50 E942

8900 18 01 B7 CB 11 21 9D 02 : 6C
8908 CD 8B 00 10 EA E1 71 23 : C7
8910 C1 10 DF E5 CD 4B 02 3E : ED
8918 FF CE 00 E1 77 21 8E 04 : D8
8920 11 96 02 CD C0 02 CD 23 : 28
8928 01 D1 D5 21 95 02 ED 4B : 97
8930 00 00 ED B0 E1 D1 C1 C9 : D9
8938 CD A3 01 18 E9 CD B5 01 : F5
8940 18 E4 00 00 00 00 00 00 : FC
8948 00 00 D5 E5 D5 11 AE 04 : 52
8950 CD C0 02 EB D1 CD E1 05 : FE
8958 CD 5D 03 EB E1 CD DB 02 : A3
8960 D1 C9 00 00 00 00 00 00 : 9A
8968 00 00 C5 D5 E5 CD C1 00 : 0D
8970 0E 01 3A 9E 02 47 3A 94 : FE
8978 02 B8 28 08 FE 01 20 02 : 0B
---------------------------------
SUM: 17 F7 5C 8D CA D0 53 40 C912

8980 0E FF 18 28 3A 9F 02 47 : 6F
8988 3A 95 02 B8 28 06 30 02 : E9
8990 0E FF 18 0D CD 4B 02 20 : 6C
8998 04 0E 00 18 04 30 02 0E : 6E
89A0 FF 3A 94 02 FE 01 20 04 : F2
89A8 79 ED 44 4F 79 3C D6 01 : 85
89B0 E1 D1 C1 C9 D5 E5 16 00 : 0C
89B8 5F 21 12 05 CD BE 06 EB : 13
89C0 E1 CD EA 02 D1 C9 00 00 : 34
89C8 00 00 00 00 00 00 D5 11 : E6
89D0 F3 09 CD 5D 03 D1 C9 D5 : 98
89D8 11 F3 09 CD EB 03 D1 C9 : 62
89E0 C5 D5 E5 CD E8 00 3E 00 : 72
89E8 32 94 02 3A 95 02 B7 CA : 1A
89F0 72 04 D6 81 1F 38 06 21 : 4B
89F8 96 02 CD 72 00 C6 81 32 : 50
---------------------------------
SUM: F6 F2 27 4A A7 9D 33 33 5040

8A00 95 02 21 96 02 11 A8 02 : 0B
8A08 CD C0 02 21 03 0A 11 96 : 64
8A10 02 CD C0 02 11 A0 02 CD : 11
8A18 C0 02 11 B8 02 CD C0 02 : 1C
8A20 0E 21 3A 00 00 FE 08 20 : 8F
8A28 02 0E 39 21 BF 02 CD 8A : 82
8A30 00 21 AF 02 CD 8A 00 21 : 4A
8A38 9D 02 CD 8B 00 21 AF 02 : C9
8A40 CD 8A 00 21 9D 02 CD 8B : 6F
8A48 00 37 21 A7 02 CD 8B 00 : 59
8A50 CD 4B 02 30 10 21 A4 02 : 21
8A58 3A 00 00 FE 08 20 03 21 : 84
8A60 A7 02 35 18 21 CD C4 01 : A9
8A68 3A 00 00 47 21 BC 02 11 : 71
8A70 A4 02 FE 08 20 06 21 BF : B2
8A78 02 11 A7 02 CB FE 37 1A : D6
---------------------------------
SUM: 2C 04 E0 7E 88 D0 1C CD F661

8A80 CE 00 12 1B 10 F9 0D 20 : 31
8A88 A2 21 B8 02 11 96 02 CD : F3
8A90 C0 02 C3 72 04 CD EB 03 : B6
8A98 C5 D5 E5 06 1F 11 04 00 : B9
8AA0 3A 00 00 FE 08 20 05 06 : 6B
8AA8 37 11 07 00 7E D6 81 4F : 73
8AB0 30 09 11 03 0A EB CD C0 : CF
8AB8 02 18 1B B8 30 18 19 78 : C6
8AC0 91 FE 09 38 07 36 00 2B : 38
8AC8 D6 08 18 F5 0E FF CB 21 : E4
8AD0 3D 20 FB 7E A1 77 E1 D1 : A0
8AD8 C1 C9 D5 11 37 06 CD C0 : 3A
8AE0 02 EB CD E4 05 EB CD DB : 36
8AE8 02 D1 C9 00 00 00 00 00 : 9C
8AF0 00 00 00 D5 11 68 06 CD : 21
8AF8 C0 02 EB CD 26 06 7E FE : 22
---------------------------------
SUM: C1 D7 17 90 2D 71 34 00 0E1F

8B00 80 F5 21 37 06 CD C0 02 : 62
8B08 EB F1 D1 C0 3A 94 02 FE : 3B
8B10 01 28 19 D5 11 FB 09 CD : F9
8B18 EA 02 D1 C9 00 00 00 00 : 86
8B20 00 00 00 00 CD E4 05 23 : D9
8B28 CB 7E 2B C8 D5 11 FB 09 : 26
8B30 CD DB 02 D1 C9 E5 23 23 : 6F
8B38 23 23 23 AF 77 23 77 23 : 4C
8B40 77 E1 C9 C5 D5 E5 3A 00 : DA
8B48 00 F5 3E 08 32 00 00 CD : 3A
8B50 E8 00 3E 05 32 00 00 CD : 2A
8B58 23 01 F1 32 00 00 D1 D5 : ED
8B60 21 95 02 01 05 00 ED B0 : 5B
8B68 AF 12 13 12 13 12 E1 D1 : BD
8B70 C1 C9 C5 D5 E5 D5 21 03 : 02
8B78 0A CD E8 00 3E 90 32 95 : 54
---------------------------------
SUM: 2E A0 24 C9 A7 B5 91 C7 DC36

8B80 02 D1 6A 63 22 96 02 C3 : 1D
8B88 72 04 CB 7A 28 E4 D5 CD : 69
8B90 15 07 CD BE 06 D1 34 35 : E7
8B98 C8 23 7E EE 80 77 2B C9 : 42
8BA0 E5 7E 23 56 23 5E 23 66 : E6
8BA8 CB 7A F5 B7 28 14 CB FA : F2
8BB0 FE 90 30 09 CB 3A CB 1B : B2
8BB8 CB 1C 3C 18 F3 CB 7C 28 : 9D
8BC0 01 13 F1 E1 C9 CD EC 06 : 6E
8BC8 C8 7A 2F 57 7B 2F 5F 13 : E4
8BD0 C9 C5 22 D2 07 1A FE 20 : C1
8BD8 20 03 13 18 F8 CD C0 07 : DA
8BE0 F5 D5 11 03 0A EB CD C0 : 60
8BE8 02 EB D1 AF 32 D1 07 0E : 85
8BF0 00 0C 0D 20 07 1A FE 2E : 86
8BF8 20 02 13 0C 1A FE 30 38 : C1
---------------------------------
SUM: 93 C6 5B B7 79 F0 76 A5 6CBE

8C00 17 FE 3A 30 13 CD 1A 05 : 7E
8C08 1A D6 30 CD 00 05 13 3A : 3F
8C10 D1 07 91 32 D1 07 18 D9 : 64
8C18 1A FE 45 28 04 FE 65 20 : 0C
8C20 2E 13 CD C0 07 F5 21 00 : EB
8C28 00 1A D6 30 38 11 FE 0A : 71
8C30 30 0D 29 44 4D 29 29 09 : 52
8C38 4F 06 00 09 13 18 EA F1 : 64
8C40 FE 01 20 04 7D ED 44 6F : 40
8C48 3A D1 07 85 32 D1 07 2A : CB
8C50 D2 07 3A D1 07 B7 28 14 : DE
8C58 FE 80 30 08 47 CD 1A 05 : E9
8C60 10 FB 18 08 ED 44 47 CD : 70
8C68 23 05 10 FB F1 FE 01 CC : EF
8C70 E2 06 C1 C9 1A FE 2D 20 : D7
8C78 04 13 3E 01 C9 FE 2B 20 : 68
---------------------------------
SUM: EA 8B C4 C3 45 9E 09 C7 FCDB

8C80 01 13 3E 00 C9 00 00 00 : 1B
8C88 C5 D5 E5 D5 11 A8 02 CD : DC
8C90 C0 02 EB D1 7E B7 20 0B : DE
8C98 3E 20 12 13 3E 30 12 13 : 16
8CA0 C3 17 09 23 CB 7E CB BE : D8
8CA8 3E 20 28 02 3E 2D 12 13 : 18
8CB0 D5 21 A8 02 11 BB 09 0E : 83
8CB8 08 3A 00 00 FE 08 20 05 : 6D
8CC0 11 7B 09 0E 10 D5 06 00 : 8E
8CC8 7E FE 81 38 0F CD B6 04 : CB
8CD0 38 08 CD EB 03 78 81 47 : 3B
8CD8 18 F3 18 0B CD 5D 03 78 : D3
8CE0 91 47 7E FE 81 38 F5 78 : 7A
8CE8 32 7A 09 01 C3 09 3A 00 : BC
8CF0 00 FE 08 20 03 01 83 09 : B6
8CF8 11 F3 09 CD B6 04 38 17 : E3
---------------------------------
SUM: 55 C2 00 08 9A BA 64 2A E728

8D00 3A 7A 09 3C 32 7A 09 7B : 29
8D08 D6 08 5F 30 01 15 79 C6 : C2
8D10 08 4F 30 01 04 18 E4 50 : D8
8D18 59 CD 5D 03 CD 3F 06 D1 : 69
8D20 CD B6 04 38 0A CD 23 05 : BE
8D28 3A 7A 09 3C 32 7A 09 D1 : 7F
8D30 3E 20 12 13 D5 06 F8 3A : 90
8D38 00 00 FE 08 20 02 06 F0 : 1E
8D40 3A 7A 09 B8 38 0A ED 44 : E8
8D48 1B 47 3E 30 12 13 10 FA : FF
8D50 3A 00 00 FE 08 20 0C 01 : 6D
8D58 9B 09 CD 1D 09 01 BB 09 : 5C
8D60 CD 1D 09 01 DB 09 CD 1D : C2
8D68 09 21 A8 02 CD 44 09 E1 : CF
8D70 06 08 3A 00 00 FE 08 20 : 6E
8D78 02 06 10 3A 7A 09 4F B8 : DC
---------------------------------
SUM: BE 04 21 3F B2 C7 87 80 BB37

8D80 30 0D D5 54 5D 1B 06 00 : E4
8D88 03 ED B0 EB D1 18 0D 46 : C7
8D90 2B 7E FE 30 20 04 0E 00 : 09
8D98 18 02 70 23 79 32 7A 09 : DB
8DA0 36 2E 1B 1A FE 30 28 FA : E9
8DA8 FE 2E 28 01 13 3A 7A 09 : 25
8DB0 6F B7 28 17 3E 45 12 13 : 0D
8DB8 7D FE 80 30 04 3E 2B 18 : B0
8DC0 05 ED 44 6F 3E 2D 12 13 : 35
8DC8 CD 66 09 AF 12 E1 D1 C1 : 70
8DD0 C9 D5 21 A8 02 11 B8 02 : 34
8DD8 CD C0 02 EB 50 59 CD E1 : D1
8DE0 05 11 B0 02 CD C0 02 EB : 42
8DE8 50 59 CD 5D 03 EB 21 A8 : 8A
8DF0 02 CD DB 02 D1 21 B8 02 : 58
8DF8 D5 CD EC 06 EB D1 01 E8 : 39
---------------------------------
SUM: 2A 77 92 0C 48 6B BE B1 B013

8E00 03 3E 2F B7 3C ED 42 30 : C2
8E08 FB 09 12 13 01 64 00 3E : CC
8E10 2F B7 3C ED 42 30 FB 09 : 85
8E18 12 13 06 0A 7D 0E 2F 0C : FB
8E20 90 30 FC 80 6F 79 12 13 : 49
8E28 7D C6 30 12 13 C9 00 B6 : 17
8E30 0E 1B C9 BF 04 00 00 B2 : 67
8E38 63 5F A9 31 A0 00 00 AF : EB
8E40 35 E6 20 F4 80 00 00 AC : 5B
8E48 11 84 E7 2A 00 00 00 A8 : 4E
8E50 68 D4 A5 10 00 00 00 A5 : 96
8E58 3A 43 B7 40 00 00 00 A2 : 16
8E60 15 02 F9 00 00 00 00 9E : AE
8E68 6E 6B 28 00 00 00 00 9B : 9C
8E70 3E BC 20 00 00 00 00 98 : B2
8E78 18 96 80 00 00 00 00 94 : C2
---------------------------------
SUM: 7E C1 45 B1 A2 D1 7E AD BC83

8E80 74 24 00 00 00 00 00 91 : 29
8E88 43 50 00 00 00 00 00 8E : 21
8E90 1C 40 00 00 00 00 00 8A : E6
8E98 7A 00 00 00 00 00 00 87 : 01
8EA0 48 00 00 00 00 00 00 84 : CC
8EA8 20 00 00 00 00 00 00 81 : A1
8EB0 00 00 00 00 00 00 00 00 : 00
8EB8 00 00 00 00 00 00 00 D5 : D5
8EC0 11 1F 0A CD C0 02 CD 32 : C8
8EC8 0A EB CD 27 0A EB CD EB : 96
8ED0 03 D1 C9 00 00 00 00 00 : 9D
8ED8 00 00 00 C5 D5 E5 11 B9 : 49
8EE0 0A CD EA 02 18 03 C5 D5 : 78
8EE8 E5 0E 00 11 A8 02 CD C0 : 3B
8EF0 02 EB 11 C9 0A CD 96 04 : 38
8EF8 23 CB 7E 28 03 CB BE 0C : 2C
---------------------------------
SUM: E7 20 19 BD 6C 6F 91 85 E21A

8F00 2B 11 C1 0A CD B6 04 38 : C6
8F08 04 CD DB 02 0C 11 B9 0A : 8E
8F10 CD B6 04 38 07 11 C1 0A : A2
8F18 CD DB 02 0C CB 41 C4 E2 : 68
8F20 06 7E FE 4D 30 0A D1 21 : FB
8F28 03 0A CD C0 02 EB 18 3A : D9
8F30 11 B0 02 CD C0 02 EB CD : 0A
8F38 5D 03 D1 21 D1 0A CD C0 : BA
8F40 02 EB 06 0A 3A 00 00 FE : 35
8F48 05 20 05 06 08 11 E1 0A : 34
8F50 D5 11 B0 02 CD 5D 03 D1 : 96
8F58 7B C6 08 5F 30 01 14 CD : BA
8F60 EA 02 10 EC 11 A8 02 CD : 70
8F68 5D 03 D1 C1 C9 81 49 0F : 94
8F70 DA A2 21 68 C2 82 49 0F : A1
8F78 DA A2 21 68 C2 83 49 0F : A2
---------------------------------
SUM: 92 D5 26 39 0B B7 B8 B6 F8F8

8F80 DA A2 21 68 C2 3F 38 DC : 1A
8F88 77 B6 E7 AB 8C 48 97 A4 : CE
8F90 DA 34 0A 0A B9 50 4A 96 : 0B
8F98 3B 81 85 6A 53 58 D7 3F : 6C
8FA0 9F 39 9D C0 F9 60 30 92 : 50
8FA8 30 9D 43 68 4C 67 D7 32 : 34
8FB0 2B 3F AA 27 1C 6E 38 EF : EC
8FB8 1D 2A B6 39 9C 74 D0 0D : 23
8FC0 00 D0 0D 00 D0 7A 08 88 : B7
8FC8 88 88 88 88 89 7E AA AA : 7B
8FD0 AA AA AA AA AB 81 00 00 : D4
8FD8 00 00 00 00 00 C5 D5 23 : BD
8FE0 CB 7E CB BE 2B F5 11 FB : FE
8FE8 09 CD B6 04 30 05 CD 73 : 05
8FF0 0B 18 2D 20 0B 11 B9 0A : 4F
8FF8 EB CD C0 02 EB 35 18 20 : D2
---------------------------------
SUM: 79 7E 84 25 AC 56 35 02 D82D

9000 11 A8 02 CD C0 02 11 FB : 56
9008 09 EB CD C0 02 EB 11 A8 : 27
9010 02 CD EB 03 CD 73 0B 11 : 19
9018 B9 0A CD DB 02 CD E2 06 : 22
9020 F1 C4 E2 06 D1 C1 C9 11 : 09
9028 E0 0B CD B6 04 F5 38 1B : BA
9030 11 A8 02 CD C0 02 11 E0 : 3B
9038 0B CD DB 02 E5 21 A8 02 : 65
9040 CD 5D 03 CD 5F 06 EB E1 : 2B
9048 CD EB 03 11 A8 02 CD C0 : 03
9050 02 11 B0 02 CD C0 02 EB : 3F
9058 CD 5D 03 21 F0 0B CD C0 : D6
9060 02 EB 06 12 3A 00 00 FE : 3D
9068 05 20 05 06 0A 11 30 0C : 87
9070 D5 11 B0 02 CD 5D 03 D1 : 96
9078 7B C6 08 5F 30 01 14 CD : BA
---------------------------------
SUM: 82 46 8F 70 10 48 97 BC 11DB

9080 EA 02 10 EC 11 A8 02 CD : 70
9088 5D 03 F1 38 06 11 E8 0B : 93
9090 CD EA 02 C9 7F 54 12 05 : 6C
9098 BC 01 A3 6E 7F 49 0E 56 : FA
90A0 32 C5 5B 5C 7B 5D 67 C8 : B5
90A8 A6 0D D6 7D 7B EA 0E A0 : 19
90B0 EA 0E A0 EA 7B 78 3E 0F : C2
90B8 83 E0 F8 3E 7C 84 21 08 : C2
90C0 42 10 84 21 7C 0D 3D CB : 88
90C8 08 D3 DC B1 7C 97 B4 25 : 54
90D0 ED 09 7B 42 7C 23 D7 0A : 33
90D8 3D 70 A3 D7 7C B2 16 42 : AD
90E0 C8 59 0B 21 7C 43 0C 30 : 48
90E8 C3 0C 30 C3 7C D7 94 35 : DE
90F0 E5 0D 79 43 7C 70 F0 F0 : 7A
90F8 F0 F0 F0 F1 7D 88 88 88 : D6
---------------------------------
SUM: E9 6E 91 5F E3 24 D4 CB 4879

9100 88 88 88 89 7D 1D 89 D8 : 1C
9108 9D 89 D8 9E 7D BA 2E 8B : 8C
9110 A2 E8 BA 2F 7D 63 8E 38 : 19
9118 E3 8E 38 E4 7E 92 49 24 : 0A
9120 92 49 24 92 7E 4C CC CC : F3
9128 CC CC CC CD 7F AA AA AA : AE
9130 AA AA AA AB 81 00 00 00 : 2A
9138 00 00 00 00 D5 E5 23 CB : A8
```

▶私は始めてOh!Xを買いました。X68000を買って約1カ月，ゲームしかノウのないこの私が，どうしてこの本を買ったのか？　それは友だちに「この本はいいよ」と勧められたからなのです。単純なやつだと笑わないでください。たぶん，毎月買うことになると思いますから……。しかし，専門用語がわからなイイイ……。　竹沢　裕利（17）千葉県

```
9140 7E CB BE 3E 00 28 02 3E : AD
9148 01 32 0E 0D 21 A8 02 EB : 04
9150 CD C0 02 EB CD 26 06 7E : F1
9158 E1 D1 B7 20 05 1A FE 91 : 37
9160 38 08 3A 0E 0D FE 01 C8 : 5C
9168 18 59 D5 C5 E5 EB CD EC : 94
9170 06 42 4B EB E1 C5 79 D6 : 73
9178 32 78 DE 00 38 05 CD 0F : A1
---------------------------------
SUM: 67 EF A9 58 46 6A 43 D1 6247

9180 0D 18 2F 13 06 00 1A CB : 52
9188 7F 28 02 06 01 11 A8 02 : 6B
9190 CD C0 02 11 FB 09 EB CD : 5C
9198 C0 02 EB 11 A8 02 0C 0D : 81
91A0 28 10 78 FE 00 20 05 CD : A0
91A8 5D 03 18 03 CD EB 03 0D : 43
91B0 18 EC C1 3A 0E 0D FE 01 : 19
91B8 20 05 CB 41 C4 E2 06 C1 : 9E
91C0 D1 C9 00 CD F6 0D CD 5D : 94
91C8 03 C5 D5 E5 11 A8 02 CD : 0A
91D0 C0 02 EB 11 66 0D CD EB : E9
91D8 03 CD 26 06 11 66 0D CD : 4D
91E0 5D 03 D1 21 6E 0D CD C0 : 5A
91E8 02 EB 06 10 3A 00 00 FE : 3B
91F0 05 20 05 06 0C 11 8E 0D : E8
91F8 D5 11 A8 02 CD 5D 03 D1 : 8E
---------------------------------
SUM: A6 82 A4 B9 48 B9 CC C1 46A0

9200 7B C6 08 5F 30 01 14 CD : BA
9208 EA 02 10 EC E5 21 37 06 : 2B
9210 CD 11 07 E1 7E 83 77 D1 : 0F
9218 C1 C9 80 31 72 17 F7 D1 : 8C
9220 CF 7A 54 57 3F 9F 39 9D : A8
9228 C0 F9 58 57 3F 9F 39 9D : 1C
9230 C0 F9 5C 49 CB A5 46 03 : 17
9238 E4 E9 60 30 92 30 9D 43 : FF
9240 68 4C 64 0F 76 C7 7F C6 : A9
9248 C4 BE 67 57 32 2B 3F AA : 86
9250 27 1C 6B 13 F2 7D BB C4 : AF
9258 FA E4 6E 38 EF 1D 2A B6 : 70
9260 39 9C 71 50 0D 00 D0 0D : 80
9268 00 D0 74 50 0D 00 D0 0D : 7E
9270 00 D0 77 36 0B 60 B6 0B : A9
```

```
9278 60 B6 7A 08 88 88 88 88 : B8
---------------------------------
SUM: 0C F3 81 13 16 43 8F 8C 0225

9280 88 89 7C 2A AA AA AA AA : 5F
9288 AA AB 7E 2A AA AA AA AA : A5
9290 AA AB 80 00 00 00 00 00 : D5
9298 00 00 81 00 00 00 00 00 : 81
92A0 00 00 81 00 00 00 00 00 : 81
92A8 00 00 C5 D5 E5 11 A8 02 : 3A
92B0 CD C0 02 EB 7E 32 70 0E : A8
92B8 36 81 11 B8 02 CD C0 02 : 11
92C0 CD 78 06 EB CD 5F 06 EB : 53
92C8 CD EB 03 11 B0 02 CD C0 : 0B
92D0 02 EB CD 5D 03 D1 21 71 : 7D
92D8 0E CD C0 02 EB 06 0E 3A : D6
92E0 00 00 FE 05 20 05 06 08 : 36
92E8 11 A1 0E D5 11 B0 02 CD : 25
92F0 5D 03 D1 7B C6 08 5F 30 : 09
92F8 01 14 CD EA 02 10 EC 11 : DB
---------------------------------
SUM: F8 F3 94 66 1D 69 81 D2 EE80

9300 A8 02 CD 5D 03 34 E5 3A : 2A
9308 70 0E D6 81 5F 7A 9A 57 : 9F
9310 21 B8 02 CD D6 06 11 66 : FB
9318 0D CD 5D 03 EB E1 CD EA : BD
9320 02 D1 C1 C9 00 7C 0D 3D : 23
9328 CB 08 D3 DC B1 7C 17 B4 : 7A
9330 25 ED 09 7B 42 7C 23 D7 : 4E
9338 0A 3D 70 A3 D7 7C 32 16 : F5
9340 42 C8 59 0B 21 7C 43 0C : 5A
9348 30 C3 0C 30 C3 7C 57 94 : 59
9350 35 E5 0D 79 43 7C 70 F0 : BF
9358 F0 F0 F0 F0 F1 7D 08 88 : BE
9360 88 88 88 88 89 7D 1D 89 : CC
9368 D8 9D 89 D8 9E 7D 3A 2E : 59
9370 8B A2 E8 BA 2F 7D 63 8E : 6C
9378 38 E3 8E 38 E4 7E 12 49 : 9E
---------------------------------
SUM: FC A2 F8 67 3F 6B B4 65 3C78

9380 24 92 49 24 92 7E 4C CC : 4B
9388 CC CC CC CC CD 7F 2A AA : 50
9390 AA AA AA AA AB 81 00 00 : D4
```

```
9398 00 00 00 00 00 23 CB BE : AC
93A0 2B C9 34 35 C8 D5 23 CB : E8
93A8 7E 2B F5 11 FB 09 EB CD : 6B
93B0 C0 02 EB F1 C4 E2 06 D1 : 1B
93B8 C9 D5 18 07 D5 11 17 0F : C9
93C0 CD EB 03 11 C1 0A CD 5D : C1
93C8 03 D1 C9 88 34 00 00 00 : 59
93D0 00 00 00 1F 0F 1F 0F 1F : 7B
93D8 0F 25 0F 25 0F 25 0F 2B : D6
93E0 0F 2B 0F 2B 0F 31 0F 31 : F4
93E8 0F 31 0F 37 0F 37 0F 37 : 12
93F0 0F 3D 0F 3D 0F 3D 0F 43 : 36
93F8 0F 43 0F 43 0F 49 0F 49 : 54
---------------------------------
SUM: E7 90 02 97 B5 AE 93 47 735E

9400 0F 49 0F 4F 0F 4F 0F 4F : 72
9408 0F 55 0F 55 0F 55 0F 5B : 96
9410 0F 5B 0F 5B 0F 61 0F 61 : B4
9418 0F 61 0F 67 0F 67 0F 67 : D2
9420 0F 6D 0F 6D 0F 6D 0F 73 : F6
9428 0F 73 0F 73 0F 79 0F 79 : 14
9430 0F 79 0F 7F 0F 7F 0F 7F : 32
9438 0F 85 0F 85 0F 85 0F 8B : 56
9440 0F 8B 0F 8B 0F 91 0F 91 : 74
9448 0F 91 0F 97 0F 97 0F 97 : 92
9450 0F 9D 0F 9D 0F 9D 0F A3 : B6
9458 0F A3 0F A3 0F A9 0F A9 : D4
9460 0F A9 0F AF 0F AF 0F AF : F2
9468 0F B5 0F B5 0F B5 0F BB : 16
9470 0F BB 0F BB 0F C1 0F C1 : 34
9478 0F C1 0F C7 0F C7 0F C7 : 52
---------------------------------
SUM: F0 6E F0 92 F0 B0 F0 CE B15A

9480 0F CD 0F CD 0F CD 0F D3 : 76
9488 0F D3 0F D3 0F D9 0F D9 : 94
9490 0F D9 0F DF 0F DF 0F DF : B2
9498 0F E5 0F E5 0F E5 0F EB : D6
94A0 0F EB 0F EB 0F F1 0F F1 : F4
94A8 0F F1 0F F7 0F F7 0F F7 : 12
94B0 0F 00 00 00             : 0F
---------------------------------
SUM: 69 3A 5A 46 5A 52 5A 5E 393E
```


リスト2 ローダ

```
3000 DD 21 00 00 DD 39 21 00 : 35
3008 00 E5 18 02 E1 C9 CD E2 : 58
3010 1F 4C 6F 61 64 20 46 69 : 6E
3018 6C 65 20 4E 61 6D 65 00 : 7F
3020 00 ED 5B 76 1F CD D3 1F : 9C
3028 1A FE 1B 28 DF 3E 01 CD : 46
3030 A3 1F CD E2 1F 4C 6F 61 : AC
3038 64 20 41 64 64 72 65 73 : D7
3040 73 0D 00 ED 5B 76 1F CD : 2A
3048 D3 1F CD B2 1F 38 BD DD : 62
3050 75 FE DD 74 FF CD 09 20 : B9
3058 38 B2 CD 80 1F 11 A4 00 : 0B
3060 19 C4 81 1F 20 EF CD E2 : 3B
3068 1F 4C 6F 61 64 69 6E 67 : DD
3070 20 00 CD 9D 1F CD EE 1F : 83
```

```
3078 DD 6E FE DD 66 FF 22 70 : 1D
---------------------------------
SUM: B1 3B 5D 22 A5 08 15 BA 9764

3080 1F CD A6 1F 38 86 2A 70 : 09
3088 1F E5 FD E1 FD 5E 00 FD : 3A
3090 23 FD 56 00 FD 23 19 EB : 9A
3098 FD 6E 00 FD 23 FD 66 00 : EE
30A0 FD 23 7D B4 28 16 19 4D : F5
30A8 44 7E 23 66 6F D5 ED 5B : D7
30B0 70 1F 19 D1 C5 E3 C1 71 : 53
30B8 23 70 18 DC FD 6E 00 FD : EF
30C0 23 FD 66 00 FD 23 7D B4 : D7
30C8 28 09 19 46 DD 7E FE 80 : 69
30D0 77 18 E9 FD 6E 00 FD 23 : 03
```

```
30D8 FD 66 00 FD 23 7D B7 28 : DF
30E0 09 19 46 DD 7E FF 80 77 : B9
30E8 18 E9 2A 70 1F 4E 23 46 : 71
30F0 2A 72 1F B7 ED 42 4D 44 : 32
30F8 EB ED 5B 70 1F ED B0 E1 : 40
---------------------------------
SUM: 27 32 1C 78 C2 DA 3F CF 70E3

3100 C9 F5 CD E2 1F 46 6F 75 : B6
3108 6E 64 20 20 00 CD 9D 1F : 9B
3110 CD EE 1F F1 C9          : 94
---------------------------------
SUM: 04 47 0C F3 E8 13 0C 94 2F26
```

リスト3 サンプルプログラム

```
A000 3E 08 32 00 B0 3E 0C CD : 3F
A008 F4 1F 21 A7 A0 11 01 00 : 8D
A010 CD 11 B0 21 A7 A0 CD 5C : 1F
A018 B0 11 CF A0 CD 17 B0 CD : 91
A020 E5 1F 21 9F A0 11 00 00 : 75
A028 CD 11 B0 21 AF A0 11 01 : 10
A030 00 CD 11 B0 21 AF A0 11 : 0F
A038 BF A0 CD 02 B0 21 C7 A0 : 66
A040 11 9D A0 CD 0E B0 21 00 : FA
A048 01 CD 1E 20 21 AF A0 11 : 8D
A050 B7 A0 CD 02 B0 62 6B CD : 70
A058 26 B0 21 C7 A0 11 A7 A0 : B6
A060 CD 02 B0 EB 11 B7 A0 CD : 9F
```

```
A068 29 B0 EB 21 9F A0 CD 20 : 11
A070 B0 11 A7 A0 CD 02 B0 21 : A8
A078 A7 A0 CD 44 B0 11 CF A0 : 88
---------------------------------
SUM: 5C 03 3C 80 90 C3 C1 D4 D50F

A080 CD 17 B0 CD E5 1F 21 AF : 35
A088 A0 11 BF A0 CD 20 B0 CD : 7A
A090 D0 1F FE 1B C2 46 A0 3E : EE
A098 0D CD F4 1F C9 36 00 00 : EC
A0A0 00 00 00 00 00 00 00 00 : 00
A0A8 00 00 00 00 00 00 00 00 : 00
A0B0 00 00 00 00 00 00 00 00 : 00
```

```
A0B8 00 00 00 00 00 00 00 00 : 00
A0C0 00 00 00 00 00 00 00 00 : 00
A0C8 00 00 00 00 00 00 00 00 : 00
A0D0 00 00 00 00 00 00 00 00 : 00
A0D8 00 00 00 00 00 00 00 00 : 00
A0E0 00 00 00 00 00 00 00 00 : 00
A0E8 00 00 00 00 00 00 00 00 : 00
A0F0 00 66 66 66 66 66 66 66 : CA
A0F8 66 66 66 66 66 66 66 66 : 30
---------------------------------
SUM: B0 E0 2D 73 09 87 3D 86 0FFC
```

リスト4 サンプルソースリスト

```
0000                  1 ;
0000                  2 ;  「マシン語版サンプル」
0000                  3 ;
0000                  4 ;         サンプル.ASM
0000                  5 ;
A000                  6         ORG $A000
A000                  7 ;
A000                  8 ;SROBAN CALL
A000                  9 ;
A000                 10 @@@      EQU $B000
A000                 11 ;
A000                 12 ;@@@は「SOROBAN」パッケージの
A000                 13 ;アドレスです。ここでは仮に$B000と
A000                 14 ;していますが、各自のシステムに合わせて
A000                 15 ;変更してください。
A000                 16 ;
A000                 17 #PRCSN  EQU @@@+$00
A000                 18 #MOVE   EQU @@@+$02
A000                 19 #CVSTF  EQU @@@+$0E
A000                 20 #CVUTF  EQU @@@+$11
A000                 21 #CVFTS  EQU @@@+$17
A000                 22 #ADD    EQU @@@+$20
A000                 23 #MUL    EQU @@@+$26
A000                 24 #DIV    EQU @@@+$29
A000                 25 #SQR    EQU @@@+$44
A000                 26 #PAI    EQU @@@+$5C
A000                 27 ;
A000                 28 ;S-OS CALL
A000                 29 ;
A000                 30 #PRINT  EQU $1FF4
A000                 31 #MSX    EQU $1FE5
```

```
A000                 32 #LOC    EQU $201E
A000                 33 #GETKY  EQU $1FD0
A000                 34 ;
A000                 35 ;倍精度
A000                 36 ;
A000 3E 08           37         LD   A,8
A002 32 00 B0        38         LD   (#PRCSN),A
A005                 39 ;
A005                 40 ;円周率パイを表示
A005                 41 ;
A005 3E 0C           42         LD   A,$0C     ;CLS
A007 CD F4 1F        43         CALL #PRINT
A00A                 44 ;
A00A 21 A7 A0        45         LD   HL,X      ;X=1
A00D 11 01 00        46         LD   DE,1
A010 CD 11 B0        47         CALL #CVUTF
A013                 48 ;
A013 21 A7 A0        49         LD   HL,X      ;X=PAI(X)
A016 CD 5C B0        50         CALL #PAI
A019                 51 ;
A019 11 CF A0        52         LD   DE,BUFF   ;PRINT X;
A01C CD 17 B0        53         CALL #CVFTS
A01F CD E5 1F        54         CALL #MSX
A022                 55 ;
A022                 56 ;円周率パイを求める
A022                 57 ;
A022 21 9F A0        58         LD   HL,P2     ;P2=0
A025 11 00 00        59         LD   DE,0
A028 CD 11 B0        60         CALL #CVUTF
A02B                 61 ;
```

▶関係ないけど,「女の60分」という番組に「トドの大和煮」が出ていた。話によるとクジラのような味だそうです。

西　薫_(18) 神奈川県

```
A02B 21 AF A0                62          LD   HL,N      ;N=1
A02E 11 01 00                63          LD   DE,1
A031 CD 11 B0                64          CALL #CVUTF
A034                         65 ;
A034 21 AF A0                66          LD   HL,N      ;CONS1=N
A037 11 BF A0                67          LD   DE,CONS1
A03A CD 02 B0                68          CALL #MOVE
A03D                         69 ;
A03D 21 C7 A0                70          LD   HL,CONS6  ;CONS6=6
A040 11 9D A0                71          LD   DE,DATA6
A043 CD 0E B0                72          CALL #CVSTF
A046                         73 ;
A046                         74 ;
A046                         75 LOOP:
A046 21 00 01                76          LD   HL,$0100  ;LOCATE(0,1)
A049 CD 1E 20                77          CALL #LOC
A04C                         78 ;
A04C 21 AF A0                79          LD   HL,N      ;N2=N
A04F 11 B7 A0                80          LD   DE,N2
A052 CD 02 B0                81          CALL #MOVE
A055                         82 ;
A055 62 6B                   83          LD   HL,DE     ;N2=N2*N2
A057 CD 26 B0                84          CALL #MUL
A05A                         85 ;
A05A 21 C7 A0                86          LD   HL,CONS6  ;X=CONS6
A05D 11 A7 A0                87          LD   DE,X
A060 CD 02 B0                88          CALL #MOVE
A063                         89 ;
A063 EB                      90          EX   DE,HL     ;X=X/N2
A064 11 B7 A0                91          LD   DE,N2
A067 CD 29 B0                92          CALL #DIV
A06A                         93 ;
A06A EB                      94          EX   DE,HL     ;P2=P2+X
A06B 21 9F A0                95          LD   HL,P2
A06E CD 20 B0                96          CALL #ADD
A071                         97 ;
A071 11 A7 A0                98          LD   DE,X      ;X=P2
A074 CD 02 B0                99          CALL #MOVE
A077                        100 ;
A077 21 A7 A0               101          LD   HL,X      ;X=SQR(X)
A07A CD 44 B0               102          CALL #SQR
```

```
A07D                         103 ;
A07D                         104 ; LD   HL,X      ;PRINT X;
A07D 11 CF A0                105       LD   DE,BUFF
A080 CD 17 B0                106       CALL #CVFTS
A083 CD E5 1F                107       CALL #MSX
A086                         108 ;
A086 21 AF A0                109       LD   HL,N      ;N=N+CONS1
A089 11 BF A0                110       LD   DE,CONS1
A08C CD 20 B0                111       CALL #ADD
A08F                         112 ;
A08F CD D0 1F                113       CALL #GETKY
A092 FE 1B C2 46 A0          114       IF   A<>$1B JP LOOP
A097                         115 ;
A097 3E 0D                   116       LD   A,$0D    ;PRINT
A099 CD F4 1F                117       CALL #PRINT
A09C C9                      118       RET
A09D                         119 ;
A09D 36 00                   120 DATA6 DM "6" DB 0
A09F                         121 ;
A09F 00 00 00 00 00 00 00    122 P2    DS 8
A0A6 00
A0A7 00 00 00 00 00 00 00    123 X     DS 8
A0AE 00
A0AF 00 00 00 00 00 00 00    124 N     DS 8
A0B6 00
A0B7 00 00 00 00 00 00 00    125 N2    DS 8
A0BE 00
A0BF 00 00 00 00 00 00 00    126 CONS1 DS 8
A0C6 00
A0C7 00 00 00 00 00 00 00    127 CONS6 DS 8
A0CE 00
A0CF                         128 ;
A0CF 00 00 00 00 00 00 00    129 BUFF  DS 34
A0D6 00 00 00 00 00 00 00
A0DD 00 00 00 00 00 00 00
A0E4 00 00 00 00 00 00 00
A0EB 00 00 00 00 00 00
A0F1                         130 ;
A0F1                         131 ; END
OBJECT CODE END A0F0
```

リスト5 SOROBANソースリスト(参考)

```
                       1 ;########################################
                       2 ;
                       3 ;     浮動小数点演算パッケージ
                       4 ;
                       5 ;       SOROBAN  2級バージョン
                       6 ;
                       7 ;         SOROBAN  0000.Asm
                       8 ;
                       9 ;########################################
                      10 ;
0000:                 11 ｱﾄﾞﾚｽ  EQU $0000
                      12 ;
0005:                 13 単精度 EQU 5
0008:                 14 倍精度 EQU 8
0008:                 15 BYTE   EQU 8
                      16 ;
0001:                 17 TRUE   EQU 1
0000:                 18 FALSE  EQU 0
0000:                 19 正     EQU 0
0001:                 20 負     EQU 1
0081:                 21 BIAS   EQU $81
                      22 ;
                      23 $IF ｱﾄﾞﾚｽ=$1234
                      24
                      25      OFFSET $9000-ｱﾄﾞﾚｽ
                      26      ORG    ｱﾄﾞﾚｽ
                      27
                      28 $ELSE
                      29
                      30      OFFSET $8000-ｱﾄﾞﾚｽ
                      31      ORG    ｱﾄﾞﾚｽ
                      32
                      33 $ENDIF
                      34 ;
                      35 ;****************************************
                      36 ;
                      37 ;     精度の型 ( 5:単精度  8:倍精度 )
                      38 ;
                      39 ;****************************************
                      40 ;
0000: 08 00           41 型  DW 倍精度
                      42 ;
                      43 ;****************************************
                      44 ;
                      45 ;     ジャンプ・テーブル
                      46 ;
                      47 ;****************************************
                      48 ;
                      49 ;転送・交換
                      50 ;
0002  C3 C0 02        51      JP [MOVE]   ;転送 [HL] --> [DE]
0005  C3 CD 02        52      JP [SWAP]   ;交換 [HL] <-> [DE]
                      53 ;
                      54 ;変換
                      55 ;
0008  C3 81 06        56      JP [CVDBL]  ;単精度 --> 倍精度
000B  C3 8F 06        57      JP [CVSNG]  ;倍精度 --> 単精度
                      58 ;
000E  C3 1D 07        59      JP [CVSTF]  ;文字列     --> 浮動小数点
0011  C3 BE 06        60      JP [CVUTF]  ;無符号整数 --> 浮動小数点
0014  C3 D6 06        61      JP [CVITF]  ;符号付整数 --> 浮動小数点
                      62 ;
0017  C3 D4 07        63      JP [CVFTS]  ;浮動小数点 --> 文字列
001A  C3 EC 06        64      JP [CVFTU]  ;浮動小数点 --> 無符号整数
001D  C3 11 07        65      JP [CVFTI]  ;浮動小数点 --> 符号付整数
                      66 ;
                      67 ;演算・関数
                      68 ;
0020  C3 EA 02        69      JP [ADD]    ;[HL]+[DE] --> [HL]
0023  C3 DB 02        70      JP [SUB]    ;[HL]-[DE] --> [HL]
0026  C3 5D 03        71      JP [MUL]    ;[HL]*[DE] --> [HL]
0029  C3 EB 03        72      JP [DIV]    ;[HL]/[DE] --> [HL]
002C  C3 E1 05        73      JP [IDIV]   ;[HL]¥[DE] --> [HL]
002F  C3 96 04        74      JP [MOD]    ;[HL]%[DE] --> [HL]
0032  C3 B6 04        75      JP [CMP]    ;[HL]-[DE] --> FLG
0035  C3 E2 06        76      JP [NEG]    ;符号反転
                      77 ;
0038  C3 70 06        78      JP [INT]    ;引数を越えない最大の整数
003B  C3 E4 05        79      JP [FIX]    ;小数点以下切り捨て
003E  C3 26 06        80      JP [FRAC]   ;引数の値の小数部
0041  C3 3F 06        81      JP [CINT]   ;小数点以下四捨五入
                      82 ;
0044  C3 2C 05        83      JP [SQR]    ;平方根
0047  C3 32 0A        84      JP [SIN]    ;三角関数ＳＩＮ
004A  C3 27 0A        85      JP [COS]    ;三角関数ＣＯＳ
004D  C3 0B 0A        86      JP [TAN]    ;三角関数ＴＡＮ
0050  C3 29 0B        87      JP [ATN]    ;逆三角関数ＡＴＮ
0053  C3 15 0D        88      JP [EXP]    ;指数関数
0056  C3 F6 0D        89      JP [LOG]    ;対数関数
                      90 ;
0059  C3 88 0C        91      JP [POW]    ;べき乗
005C  C3 05 0F        92      JP [PAI]    ;円周率の引数倍
005F  C3 08 0F        93      JP [RAD]    ;ラジアン単位に変換
0062  C3 E9 0E        94      JP [ABS]    ;絶対値
0065  C3 EE 0E        95      JP [SGN]    ;符号
                      96 ;
0068  C3 68 00        97      JP $
006B  C3 6B 00        98      JP $
006E  C3 6E 00        99      JP $
                     100 ;
                     101 ;****************************************
                     102 ;
                     103 ;     マクロ定義
                     104 ;
```

```
                     105 ;****************************************
                     106 ;
                     107 [>>] MACRO
                     108             HL=%1  CALL SHTR
                     109             ENDM
                     110 ;
                     111 [0>>] MACRO
                     112             HL=%1  CALL RCF.SHTR
                     113             ENDM
                     114 ;
                     115 [<<] MACRO
                     116             HL=%1+8-1  CALL SHTL
                     117             ENDM
                     118 ;
                     119 [<<0] MACRO
                     120             HL=%1+8-1  CALL RCF.SHTL
                     121             ENDM
                     122 ;
                     123 [<<1] MACRO
                     124             SCF HL=%1+8-1  CALL SHTL
                     125             ENDM
                     126 ;
                     127 ;
                     128 [BYTE>>] MACRO
                     129             HL=%1  CALL BSHTR
                     130             ENDM
                     131 ;
                     132 ;****************************************
                     133 ;
                     134 ;     サブルーチン
                     135 ;
                     136 ;****************************************
                     137 ;
0071:                138 RCF.SHTR
0071  B7             139      RCF
                     140 ;    CALL SHTR
                     141 ;    RET
                     142 ;
0072:                143 SHTR ;HL破壊
                     144 ;    JR SHTR5         ;$22,$07
0072  CB 1E 23       145      RR (HL)  INC HL ;$CB $1E
0075  CB 1E 23       146      RR (HL)  INC HL
0078  CB 1E 23       147      RR (HL)  INC HL
007B:                148 SHTR5
007B  CB 1E 23       149      RR (HL)  INC HL
007E  CB 1E 23       150      RR (HL)  INC HL
0081  CB 1E 23       151      RR (HL)  INC HL
0084  CB 1E 23       152      RR (HL)  INC HL
0087  CB 1E          153      RR (HL)
0089  C9             154      RET
                     155 ;
                     156 ;
008A:                157 RCF.SHTL
008A  B7             158      RCF
                     159 ;    CALL SHTL
                     160 ;    RET
                     161 ;
008B:                162 SHTL ;HL破壊
                     163 ;    JR SHTL5         ;$18,$16
008B  CB 16 2B       164      RL (HL)  DEC HL ;$CB,$16
008E  CB 16 2B       165      RL (HL)  DEC HL
0091  CB 16 2B       166      RL (HL)  DEC HL
0094  CB 16 2B       167      RL (HL)  DEC HL
0097  CB 16 2B       168      RL (HL)  DEC HL
009A  CB 16 2B       169      RL (HL)  DEC HL
009D  CB 16 2B       170      RL (HL)  DEC HL
00A0  CB 16          171      RL (HL)
00A2  C9             172      RET
                     173 ;
00A3:                174 SHTL5
00A3  2B 2B 2B       175      DEC HL/HL/HL
                     176 ;
00A6  CB 16 2B       177      RL (HL)  DEC HL
00A9  CB 16 2B       178      RL (HL)  DEC HL
00AC  CB 16 2B       179      RL (HL)  DEC HL
00AF  CB 16 2B       180      RL (HL)  DEC HL
00B2  CB 16          181      RL (HL)
00B4  C9             182      RET
                     183 ;
                     184 ;
00B5:                185 BSHTR
00B5  01 07 00       186      BC=7  ;パッチ
                     187 ;    BC=4
00B8  09 54 5D 2B ED B8 AF  188  ADD HL,BC  DE=HL  DEC HL  LDDR  (DE)=0
00BF  12
00C0  C9             189      RET
                     190 ;
                     191 ;
00C1:                192 [TOBUFF]
                     193 ;
00C1  D5             194      PUSH DE
                     195 ;
00C2  CD E8 00       196      [TOBUFF1]
                     197 ;
00C5  E1             198      POP  HL
                     199 ;
00C6  11 9F 02 ED 4B 00 00  200  DE=指数2  BC=(型)  LDIR  (DE)=0
00CD  ED B0 AF 12
                     201 ;
00D1  21 A0 02       202      HL=仮数2+0
                     203 ;
00D4  3E 00 CB 7E 28 02 3E  204  A=正  IF BIT(7,(HL))=1 THEN  A=負
00DB  01
                     205 ;
```

▶このまえ友人が「あと40万円たまるとX68000が買える」なんてことを言っていた。彼はきっと長生きするだろう。　三戸　詳司 (18) 京都府

```
00DC  32 9E 02              206        (符号2)=A
                            207 ;
00DF  3A 9F 02 B7 28 02 CB  208        IF (指数2)<>0 THEN  SET 7,(HL)
00E6  FE
00E7  C9                    209        RET
                            210 ;
                            211 ;
00E8:                       212 [TOBUFF1]
                            213 ;
00E8  ED 4B 00 00           214        BC=(型)
                            215 ;
                            216 ;パッチ
                            217 ;
00EC  79                    218        A=C
00ED  FE 08 20 05           219        IF A=倍精度 THEN
00F1  11 CB 1E              220          DE=$1ECB ;RR (HL)
                            221 ;        DE=$16CB ;RL (HL)
                            222 ;        A=8
00F4  18 03                 223        ELSE
00F6  11 18 07              224          DE=$0718 ;JR SHTR5
                            225 ;        DE=$1618 ;JR SHTL5
                            226 ;        A=5
                            227        FI
00F9  3D 32 B6 00           228        DEC A  (BSHTR+1)=A
00FD  ED 53 72 00           229        (SHTR)=DE
0101  7B 32 8B 00           230        (SHTL)=E
                            231 ;
                            232 ;
                            233 ;      BC=(型)
0105  11 95 02 ED B0 AF 12  234        DE=指数1  LDIR  (DE)=0
                            235 ;
010C  21 96 02              236        HL=仮数1+0
                            237 ;
010F  3E 00 CB 7E 28 02 3E  238        A=正  IF BIT(7,(HL))=1 THEN  A=負
0116  01
                            239 ;
0117  32 94 02              240        (符号1)=A
                            241 ;
011A  3A 95 02 B7 28 02 CB  242        IF (指数1)<>0 THEN  SET 7,(HL)
0121  FE
0122  C9                    243        RET
                            244 ;
                            245 ;
0123:                       246 [正規化]
                            247 ;0
0123  3A 95 02 B7 C8        248        IF (指数1)=0  RET
                            249 ;
                            250 ;左へシフト
                            251 ;
0128  21 96 02 3A 95 02 47  252        HL=仮数1+0  B=(指数1)  C=(型)  DE=8
012F  3A 00 00 4F 11 08 00
                            253 ;      D=0
                            254 ;      E=8
                            255        {
0136  7E B7 20 0E           256          A=(HL)  IF A<>0 EXIT ;WHILE (HL)=0 {
013A  14 23                 257          INC D/HL
013C  78 93 47 28 62        258          SUB B,E IF Z JR [0値]
0141  38 60                 259                  IF C JR [0値]
0143  0D 28 5D              260          DEC C   IF Z JR [0値]
0146  18 EE                 261        }
0148  78 32 95 02           262        (指数1)=B
                            263 ;
014C  7A                    264        A=D
014D  B7 28 0D              265        IF A<>0 THEN
0150  11 96 02 06 00 ED B0  266          DE=仮数1+0  B=0  LDIR
                            267 ;
0157  47 AF 12 13 10 FB     268          DO B,A { (DE)=0  INC DE }
                            269        FI
                            270 ;
015D  11 96 02 3A 95 02 47  271        DE=仮数1+0  B=(指数1)
                            272 ;
0164  1A CB 7F 20 0B        273        UNTIL BIT(7,(DE))=1 {
                            274          [<<0] 仮数1
0169  21 9D 02 CD 8A 00     274+              HL=仮数1+8-1  CALL RCF.SHTL
016F  05 28 31              275          DEC B  IF Z JR [0値]
0172  18 F0                 276        }
0174  78 32 95 02           277        (指数1)=B
                            278 ;
                            279 ;0捨1入
                            280 ;
0178  3A 00 00              281        A=(型)
017B  21 9A 02 FE 08 20 03  282        HL=仮数1+4 IF A=倍精度 THEN HL=仮数1+7
0182  21 9D 02
                            283 ;
0185  CB 7E 28 0D           284        IF BIT(7,(HL))=1 THEN
                            285 ;        A=(型)
0189  3D                    286          DEC A
018A  47                    287          B=A
018B  37 2B 7E CE 00 77 10  288          SCF  DO B { DEC HL  ADC (HL),0 }
0192  F9
                            289 ;
0193  DC 3C 02              290          IF C [BITあふれ]
                            291        FI
                            292 ;
                            293 ;符号
                            294 ;
0196  21 96 02 3A 94 02 FE  295        HL=仮数1+0 IF (符号1)=正 THEN RES 7,(HL)
019D  00 20 02 CB BE
                            296 ;
01A2  C9                    297        RET
                            298 ;
                            299 ;
01A3:                       300 [0値]
01A3  3E 00 32 94 02        301        (符号1)=正
01A8  AF 32 95 02           302        (指数1)=0
01AC  21 03 0A 11 96 02 C3  303        HL=値0  DE=仮数1  [LDIR]  RET
01B3  C0 02
                            304 ;
                            305 ;
01B5:                       306 [MAX値]
                            307 ;      (符号1)=?
01B5  21 95 02              308        HL=指数1
                            309 ;
01B8  3A 00 00 36 FF 23 3D  310        DO A,(型) { (HL)=$FF  INC HL }  (HL)=0
01BF  20 FA 36 00
                            311 ;
01C3  C9                    312        RET
                            313 ;
                            314 ;
01C4:                       315 [仮数減算] ;DE/HL破壊
                            316 ;
                            317 ;仮数1 = 仮数1 - 仮数2
                            318 ;
                            319 ;      ただし 仮数1 > 仮数2
                            320 ;
01C4  3A 00 00 FE 05        321        A=(型) CP 単精度
                            322 ;
01C9  20 08                 323        IF Z THEN
                            324 ;        RCF
01CB  11 9A 02 21 A4 02     325          DE=仮数1+4  HL=仮数2+4
                            326 ;
01D1  18 15                 327        ELSE
01D3  11 9D 02 21 A7 02     328          DE=仮数1+7  HL=仮数2+7
                            329 ;
01D9  1A 96 12 1B 2B        330          SUB (DE),(HL)  DEC DE/HL
01DE  1A 9E 12 1B 2B        331          SBC (DE),(HL)  DEC DE/HL
01E3  1A 9E 12 1B 2B        332          SBC (DE),(HL)  DEC DE/HL
                            333        FI
                            334 ;
01E8  1A 9E 12 1B 2B        335        SBC (DE),(HL)  DEC DE/HL
01ED  1A 9E 12 1B 2B        336        SBC (DE),(HL)  DEC DE/HL
01F2  1A 9E 12 1B 2B        337        SBC (DE),(HL)  DEC DE/HL
01F7  1A 9E 12 1B 2B        338        SBC (DE),(HL)  DEC DE/HL
01FC  1A 9E 12              339        SBC (DE),(HL)
                            340 ;
01FF  C9                    341        RET
                            342 ;
                            343 ;
0200:                       344 [仮数加算] ;DE/HL破壊
                            345 ;
                            346 ;仮数1 = 仮数1 + 仮数2
                            347 ;
0200  3A 00 00 FE 05        348        A=(型) CP 単精度
                            349 ;
0205  20 08                 350        IF Z THEN
                            351 ;        RCF
0207  11 9A 02 21 A4 02     352          DE=仮数1+4  HL=仮数2+4
                            353 ;
020D  18 15                 354        ELSE
020F  11 9D 02 21 A7 02     355          DE=仮数1+7  HL=仮数2+7
                            356 ;
0215  1A 86 12 1B 2B        357          ADD (DE),(HL)  DEC DE/HL
021A  1A 8E 12 1B 2B        358          ADC (DE),(HL)  DEC DE/HL
021F  1A 8E 12 1B 2B        359          ADC (DE),(HL)  DEC DE/HL
                            360        FI
                            361 ;
0224  1A 8E 12 1B 2B        362        ADC (DE),(HL)  DEC DE/HL
0229  1A 8E 12 1B 2B        363        ADC (DE),(HL)  DEC DE/HL
022E  1A 8E 12 1B 2B        364        ADC (DE),(HL)  DEC DE/HL
0233  1A 8E 12 1B 2B        365        ADC (DE),(HL)  DEC DE/HL
0238  1A 8E 12              366        ADC (DE),(HL)
                            367 ;
023B  D0                    368        IF NC RET
                            369 ;
                            370 ;      [BITあふれ]
                            371 ;      RET
                            372 ;
023C:                       373 [BITあふれ]
                            374 ;      SCF
                            375        [>>] 仮数1
023C  21 96 02 CD 72 00     375+            HL=仮数1  CALL SHTR
                            376 ;
0242  21 95 02 34 CC B5 01  377        HL=指数1  INC (HL)  IF Z [MAX値]
0249  37                    378        SCF
024A  C9                    379        RET
                            380 ;
                            381 ;
024B:                       382 [仮数比較] ;DE/HL破壊
                            383 ;
024B  11 96 02 21 A0 02     384        DE=仮数1  HL=仮数2
                            385 ;
0251  3A 00 00 FE 08 20 0F  386        IF (型)=倍精度 THEN
0258  1A BE C0 13 23        387          CP (DE),(HL)  IF NZ RET  INC DE/HL
025D  1A BE C0 13 23        388          CP (DE),(HL)  IF NZ RET  INC DE/HL
0262  1A BE C0 13 23        389          CP (DE),(HL)  IF NZ RET  INC DE/HL
                            390        FI
                            391 ;
0267  1A BE C0 13 23        392        CP (DE),(HL)  IF NZ RET  INC DE/HL
026C  1A BE C0 13 23        393        CP (DE),(HL)  IF NZ RET  INC DE/HL
0271  1A BE C0 13 23        394        CP (DE),(HL)  IF NZ RET  INC DE/HL
0276  1A BE C0 13 23        395        CP (DE),(HL)  IF NZ RET  INC DE/HL
027B  1A BE                 396        CP (DE),(HL)
027D  C9                    397        RET
                            398 ;
                            399 ;
027E:                       400 [交換] ;BC/DE/HL破壊
                            401 ;
027E  21 94 02              402        HL=符号1
0281  11 9E 02              403        DE=符号2
0284  3A 00 00 47 04 04     404        B=(型)  INC B/B
                            405 ;      [交換処理]
                            406 ;      RET
                            407 ;
028A:                       408 [交換処理]
                            409 ;
                            410        DO B {
028A  4E 1A 77 79 12        411          C=(HL)  (HL)=(DE)  (DE)=C
                            412 ;
028F  23 13                 413          INC HL/DE
0291  10 F7                 414        }
                            415 ;
0293  C9                    416        RET
                            417 ;
                            418 ;
0294: 00                    419 符号1  DS 1
0295: 00                    420 指数1  DS 1
0296: 00 00 00 00 00 00 00  421 仮数1  DS BYTE
029D  00
                            422 ;
029E: 00                    423 符号2  DS 1
029F: 00                    424 指数2  DS 1
02A0: 00 00 00 00 00 00 00  425 仮数2  DS BYTE
02A7  00
                            426 ;
02A8: 00 00 00 00 00 00 00  427 VAR.X   DS BYTE
02AF  00
02B0: 00 00 00 00 00 00 00  428 VAR.XX  DS BYTE
02B7  00
02B8: 00 00 00 00 00 00 00  429 VAR.Y   DS BYTE
02BF  00
                            430 ;
                            431 ;****************************************
                            432 ;
                            433 ;      転送
                            434 ;
                            435 ;        [HL] --> [DE]
                            436 ;
                            437 ;        保存 BC/DE/HL
                            438 ;
                            439 ;****************************************
                            440 ;
02C0:                       441 [MOVE]
                            442 ;
02C0:                       443 [LDIR]
                            444 ;
02C0  C5 D5 E5              445        PUSH BC/DE/HL
                            446 ;
02C3  ED 4B 00 00 ED B0     447        BC=(型)  LDIR
                            448 ;
02C9  E1 D1 C1              449        POP  HL/DE/BC
02CC  C9                    450        RET
                            451 ;
                            452 ;****************************************
                            453 ;
                            454 ;      交換
                            455 ;
                            456 ;        [HL] <-> [DE]
                            457 ;
                            458 ;        保存 BC/DE/HL
                            459 ;
                            460 ;****************************************
                            461 ;
02CD:                       462 [SWAP]
                            463 ;
02CD  C5 D5 E5              464        PUSH BC/DE/HL
                            465 ;
02D0  3A 00 00 47 CD 8A 02  466        B=(型)  [交換処理]
                            467 ;
02D7  E1 D1 C1              468        POP  HL/DE/BC
02DA  C9                    469        RET
                            470 ;
                            471 ;****************************************
                            472 ;
                            473 ;      引き算
                            474 ;
                            475 ;        [HL] = [HL] - [DE]
                            476 ;
                            477 ;        保存 BC/DE/HL
                            478 ;
                            479 ;****************************************
                            480 ;
02DB:                       481 [SUB]
                            482 ;
02DB  C5 D5 E5              483        PUSH BC/DE/HL
                            484 ;
02DE  CD C1 00              485        [TOBUFF]
                            486 ;
                            487 ;符号反転
                            488 ;
02E1  21 9E 02 7E EE 01 77  489        HL=符号2  XOR (HL),$01
                            490 ;
02E8  18 06                 491        JR [ADD処理]
                            492 ;
                            493 ;****************************************
```


▶はあはあ……，やっと2月号を見つけたぞ。テーブルの下に隠れるとはな。今度は僕が隠れるから，100数えたら探しにこいよ（居間で2月号を読んでそのあと……いつの間にか消えてしまい，1週間以上探していました）。　村上　淳一（17）福岡県

```
                              494 ;
                              495 ;     足し算
                              496 ;
                              497 ;       [HL] = [HL] + [DE]
                              498 ;
                              499 ;       保存 BC/DE/HL
                              500 ;
                              501 ;****************************************
                              502 ;
02EA:                         503 [ADD]
                              504 ;
02EA  C5 D5 E5                505       PUSH BC/DE/HL
                              506 ;
02ED  CD C1 00                507       [TOBUFF]
                              508 ;
02F0:                         509 [ADD処理]
                              510 ;
02F0  3A 95 02 47 3A 9F 02    511       B=(指数1)  C=(指数2)
02F7  4F
                              512 ;
                              513 ;指数1 >= 指数2 にする
                              514 ;
02F8  78 B9 30 08             515       IF B<C THEN
02FC  C5 CD 7E 02 C1 78 41    516         PUSH BC [交換] POP BC  EX B,C
0303  4F
                              517       FI
                              518 ;
                              519 ;ゼロ ?
                              520 ;
0304  0C 0D CA 72 04          521       IF C=0 JP [演算終了]
                              522 ;
                              523 ;小数点を合わせる
                              524 ;
0309  16 28 3A 00 00 FE 08    525       D=8*5  IF (型)=倍精度 THEN  D=8*8
0310  20 02 16 40
                              526 ;
0314  78 91 47 BA D2 72 04    527       A=B SUB C B=A  IF A>=D JP [演算終了]
031B  0E 08                   528       C=8
                              529       {
031D  78 91 38 0B             530         A=B SUB C  IF < EXIT
0321  47                      531         B=A
                              532 ;
0322  C5                      533         PUSH BC
                              534         [BYTE>>] 仮数2
0323  21 A0 02 CD B5 00       534+              HL=仮数2  CALL BSHTR
0329  C1                      535         POP  BC
032A  18 F1                   536       }
                              537 ;
032C  04 05 28 08             538       IF B<>0 THEN
                              539         DO B {
                              540           [0>>] 仮数2
0330  21 A0 02 CD 71 00       540+              HL=仮数2  CALL RCF.SHTR
0336  10 F8                   541         }
                              542       FI
                              543 ;
                              544 ;仮数の計算
                              545 ;
0338  3A 94 02 47 3A 9E 02    546       B=(符号1)  A=(符号2)
                              547 ;
033F  B8 20 05                548       IF A=B THEN
0342  CD 00 02                549         [仮数加算]
                              550 ;
0345  18 13                   551       ELSE
0347  3A 95 02 47 3A 9F 02    552         B=(指数1) A=(指数2)
                              553 ;
034E  B8 20 06                554         IF A=B THEN
0351  CD 4B 02 DC 7E 02       555           [仮数比較]  IF < [交換]
                              556         FI
                              557 ;
0357  CD C4 01                558         [仮数減算]
                              559       FI
                              560 ;
035A  C3 72 04                561       JP [演算終了]
                              562 ;
                              563 ;****************************************
                              564 ;
                              565 ;     掛け算
                              566 ;
                              567 ;       [HL] = [HL] * [DE]
                              568 ;
                              569 ;       保存 BC/DE/HL
                              570 ;
                              571 ;****************************************
                              572 ;
035D:                         573 [MUL]
                              574 ;
035D  C5 D5 E5                575       PUSH BC/DE/HL
                              576 ;
0360  CD C1 00                577       [TOBUFF]
                              578 ;
                              579 ;ゼロ ?
                              580 ;
0363  3A 95 02 B7 CA 72 04    581       IF (指数1)=0 JP [演算終了]
036A  3A 9F 02 B7 CA 84 04    582       IF (指数2)=0 JP [演算終了0値]
                              583 ;
                              584 ;符号
                              585 ;
0371  3A 9E 02 4F 3A 94 02    586       C=(符号2)  XOR (符号1),C
0378  A9 32 94 02
                              587 ;
                              588 ;指数加算
                              589 ;
037C  26 00 3A 95 02 6F       590       H=0 L=(指数1)
0382  54 3A 9F 02 5F 19       591       D=H E=(指数2) ADD HL,DE
0388  11 7F FF 19             592       DE=-BIAS      ADD HL,DE
038C  7C                      593       A=H
038D  FE FF CA 84 04          594       IF A=-1 JP [演算終了0値]
0392  FE 01 CA 89 04          595       IF A=1  JP [演算終了MAX値]
0397  7D 32 95 02             596       (指数1)=L
                              597 ;
                              598 ;仮数乗算
                              599 ;
039B  21 96 02 11 E3 03 CD    600       HL=仮数1  DE=乗数 [LDIR] ;乗数 =仮数1
03A2  C0 02
03A4  EB 21 03 0A CD C0 02    601       EX DE,HL  HL=値0  [LDIR] ;仮数1=0
                              602 ;
                              603 ;仮数1：答
                              604 ;仮数2：被乗数（本当は乗数）
                              605 ;乗数 ：乗数  （本当は被乗数）
                              606 ;
03AB  21 E3 03 3A 00 00 3D    607       HL=乗数  A=(型) DEC A ;4 or 7
                              608 ;
03B2  47                      609       DO B,A {
03B3  4E 23                   610         C=(HL)  INC HL
                              611 ;
03B5  C5 E5                   612         PUSH BC/HL
                              613 ;
03B7  0C 0D 20 08             614         IF C=0 THEN
                              615           [BYTE>>] 仮数2
03BB  21 A0 02 CD B5 00       615+              HL=仮数2  CALL BSHTR
                              616 ;
03C1  18 19                   617         ELSE
03C3  06 08                   618           DO B,8 {
03C5  CB 21                   619             SLA C
03C7  30 0B                   620             IF C THEN
03C9  CD 00 02                621               [仮数加算]
03CC  30 06                   622               IF C THEN  [0>>] 仮数2
03CE  21 A0 02 CD 71 00       622+              HL=仮数2  CALL RCF.SHTR
                              623             FI
                              624             [0>>] 仮数2
03D4  21 A0 02 CD 71 00       624+              HL=仮数2  CALL RCF.SHTR
03DA  10 E9                   625           }
                              626         FI
                              627 ;
03DC  E1 C1                   628         POP HL/BC
03DE  10 D3                   629       }
                              630 ;
03E0  C3 72 04                631       JP [演算終了]
                              632 ;
03E3: 00 00 00 00 00 00 00    633 乗数  DS BYTE
03EA  00
                              634 ;
                              635 ;****************************************
                              636 ;
                              637 ;     割り算
                              638 ;
                              639 ;       [HL] = [HL] / [DE]
                              640 ;
                              641 ;       保存 BC/DE/HL
                              642 ;
                              643 ;****************************************
                              644 ;
03EB:                         645 [DIV]
                              646 ;
03EB  C5 D5 E5                647       PUSH BC/DE/HL
                              648 ;
03EE  CD C1 00                649       [TOBUFF]
                              650 ;
                              651 ;ゼロ ?
                              652 ;
03F1  3A 95 02 B7 CA 72 04    653       IF (指数1)=0 JP [演算終了]
03F8  3A 9F 02 B7 CA 84 04    654       IF (指数2)=0 JP [演算終了0値]
                              655 ;
                              656 ;符号
                              657 ;
03FF  3A 9E 02 4F 3A 94 02    658       C=(符号2)  XOR (符号1),C
0406  A9 32 94 02
                              659 ;
                              660 ;指数減算
                              661 ;
040A  26 00 3A 95 02 6F       662       H=0 L=(指数1)
0410  54 3A 9F 02 5F B7 ED    663       D=H E=(指数2)  SUB HL,DE
0417  52
                              664 ;     D=0
0418  1E 81 19                665       E=BIAS         ADD HL,DE
041B  7C                      666       A=H
041C  FE FF CA 84 04          667       IF A=-1 JP [演算終了0値]
0421  FE 01 CA 89 04          668       IF A=1  JP [演算終了MAX値]
0426  7D 32 95 02             669       (指数1)=L
                              670 ;
                              671 ;仮数除算
                              672 ;
                              673       [0>>] 仮数1
042A  21 96 02 CD 71 00       673+              HL=仮数1  CALL RCF.SHTR
                              674       [0>>] 仮数2
0430  21 A0 02 CD 71 00       674+              HL=仮数2  CALL RCF.SHTR
                              675 ;
0436  21 8E 04 3A 00 00 3D    676       HL=商  A=(型) DEC A
                              677 ;
043D  47                      678       DO B,A {
043E  C5                      679         PUSH BC
043F  E5                      680         PUSH HL
0440  01 00 08                681         BC=$0800
                              682 ;       C=0
                              683 ;       B=8
                              684         DO B {
0443  CD 4B 02                685           [仮数比較]
0446  38 06 CD C4 01 37       686           IF >= : [仮数減算] SCF
044C  18 01 B7                687           ELSE  :            RCF
                              688           FI
044F  CB 11                   689           RL C
                              690 ;         RCF
                              691           [<<] 仮数1
0451  21 9D 02 CD 8B 00       691+              HL=仮数1+8-1  CALL SHTL
0457  10 EA                   692         }
0459  E1 71 23                693         POP HL  (HL)=C  INC HL
045C  C1                      694         POP BC
045D  10 DF                   695       }
045F  E5                      696       PUSH HL
0460  CD 4B 02 3E FF CE 00    697       [仮数比較] A=$FF ADC A,0 ;IF < THEN A=0
0467  E1                      698       POP  HL
0468  77                      699       (HL)=A
                              700 ;
0469  21 8E 04 11 96 02 CD    701       HL=商  DE=仮数1  [LDIR]
0470  C0 02
                              702 ;
                              703 ;     JP [演算終了]
                              704 ;
0472:                         705 [演算終了]
                              706 ;
0472  CD 23 01                707       [正規化]
                              708 ;
0475  D1                      709       POP  DE
0476  D5                      710       PUSH DE
                              711 ;
0477  21 95 02 ED 4B 00 00    712       HL=指数1  BC=(型)  LDIR
047E  ED B0
                              713 ;
0480  E1 D1 C1                714       POP HL/DE/BC
0483  C9                      715       RET
                              716 ;
0484:                         717 [演算終了0値]
0484  CD A3 01 18 E9          718       [0値]   JR [演算終了]
                              719 ;
0489:                         720 [演算終了MAX値]
0489  CD B5 01 18 E4          721       [MAX値] JR [演算終了]
                              722 ;
048E: 00 00 00 00 00 00 00    723 商  DS BYTE
0495  00
                              724 ;
                              725 ;****************************************
                              726 ;
                              727 ;     剰余算
                              728 ;
                              729 ;       [HL] = [HL] MOD [DE]
                              730 ;
                              731 ;       保存 BC/DE/HL
                              732 ;
                              733 ;****************************************
                              734 ;
0496:                         735 [MOD]
                              736 ;
0496  D5                      737       PUSH DE
                              738 ;
0497  E5                      739       PUSH HL
0498  D5                      740       PUSH DE
0499  11 AE 04 CD C0 02 EB    741       DE=MOD.X [MOVE]  EX DE,HL
                              742 ;     HL=MOD.X
04A0  D1 CD E1 05 CD 5D 03    743       POP  DE  [IDIV]  [MUL]  EX DE,HL
04A7  EB
                              744 ;     DE=MOD.X
04A8  E1 CD DB 02             745       POP  HL  [SUB]
                              746 ;
04AC  D1                      747       POP  DE
04AD  C9                      748       RET
                              749 ;
04AE: 00 00 00 00 00 00 00    750 MOD.X DS BYTE
04B5  00
                              751 ;
                              752 ;****************************************
                              753 ;
                              754 ;     比較
                              755 ;
                              756 ;       [HL] > [DE]  -->  NZ  NC  A=1
                              757 ;       [HL] = [DE]  -->  Z   NC  A=0
                              758 ;       [HL] < [DE]  -->  NZ  C   A=-1
                              759 ;
                              760 ;       保存 BC/DE/HL
                              761 ;
                              762 ;****************************************
                              763 ;
04B6:                         764 [CMP]
                              765 ;
04B6  C5 D5 E5                766       PUSH BC/DE/HL
                              767 ;
04B9  CD C1 00                768       [TOBUFF]
                              769 ;
04BC  0E 01                   770       C=1 ;1 > 2
                              771 ;
                              772 ;符号比較
                              773 ;
04BE  3A 9E 02 47 3A 94 02    774       B=(符号2)  A=(符号1)
                              775 ;
04C5  B8 28 08                776       IF A<>B THEN
```

▶TRONのどこがいけないんです？　私はブラインドタイプできますが，あのキーボードのほうが打ちやすそうに見えますよ。　　江原　忠士（18）岡山県

```
04C8  FE 01 20 02 0E FF      777         IF A=負 : C=-1  ;負 < 正
                             778 ;       ELSE    : C=1   ;正 > 負
                             779         FI
                             780 ;
                             781 ;指数比較
                             782 ;
04CE  18 28                  783       ELSE
04D0  3A 9F 02 47 3A 95 02   784         B=(指数2)  A=(指数1)
                             785 ;
04D7  B8 28 06               786         IF A<>B THEN
04DA  30 02 0E FF            787           IF < : C=-1   ;1 < 2
                             788 ;         ELSE : C=1    ;1 > 2
                             789           FI
                             790 ;
                             791 ;仮数比較
                             792 ;
04DE  18 0D                  793         ELSE
04E0  CD 4B 02               794           [仮数比較]
                             795 ;
04E3  20 04 0E 00            796           IF = : C=0    ;1 = 2
04E7  18 04 30 02 0E FF      797           EF < : C=-1   ;1 < 2
                             798 ;         ELSE : C=1    ;1 > 2
                             799           FI
                             800         FI
                             801 ;
04ED  3A 94 02 FE 01 20 04   802         IF (符号1)=負 THEN  A=C  NEG  C=A
04F4  79 ED 44 4F
                             803       FI
                             804 ;
04F8  79 3C D6 01            805       A=C  INC A  SUB A,1
                             806 ;
                             807 ;> : NZ  NC  A=1
                             808 ;= : Z   NC  A=0
                             809 ;< : NZ  C   A=-1
                             810 ;
04FC  E1 D1 C1               811       POP HL/DE/BC
04FF  C9                     812       RET
                             813 ;
                             814 ;****************************************
                             815 ;
                             816 ;     [HL] = [HL] + A
                             817 ;
                             818 ;     保存 BC/DE/HL
                             819 ;
                             820 ;****************************************
                             821 ;
0500:                        822 [ADDA]
                             823 ;
0500  D5                     824       PUSH DE
0501  E5                     825       PUSH HL
                             826 ;
0502  16 00 5F 21 12 05 CD   827       D=0 E=A  HL=ADDDEC.X  [CVUTF] EX DE,HL
0509  3E 06 EB
                             828 ;
050C  E1 CD EA 02            829       POP  HL  [ADD]
0510  D1                     830       POP  DE
0511  C9                     831       RET
                             832 ;
0512: 00 00 00 00 00 00 00   833 ADDDEC.X DS BYTE
0519  00
                             834 ;
                             835 ;****************************************
                             836 ;
                             837 ;     [HL] = [HL] * 10
                             838 ;
                             839 ;     保存 BC/DE/HL
                             840 ;
                             841 ;****************************************
                             842 ;
051A:                        843 [MUL10]
                             844 ;
051A  D5                     845       PUSH DE
                             846 ;
051B  11 F3 09 CD 5D 03      847       DE=値10 [MUL]
                             848 ;
0521  D1                     849       POP  DE
0522  C9                     850       RET
                             851 ;
                             852 ;****************************************
                             853 ;
                             854 ;     [HL] = [HL] / 10
                             855 ;
                             856 ;     保存 BC/DE/HL
                             857 ;
                             858 ;****************************************
                             859 ;
0523:                        860 [DIV10]
                             861 ;
0523  D5                     862       PUSH DE
                             863 ;
0524  11 F3 09 CD EB 03      864       DE=値10 [DIV]
                             865 ;
052A  D1                     866       POP  DE
052B  C9                     867       RET
                             868 ;
                             869 ;****************************************
                             870 ;
                             871 ;     平方根
                             872 ;
                             873 ;       [HL] = SQR [HL]
                             874 ;
                             875 ;       保存 BC/DE/HL
                             876 ;
                             877 ;****************************************
                             878 ;
052C:                        879 [SQR]
                             880 ;
052C  C5 D5 E5               881       PUSH BC/DE/HL
                             882 ;
052F  CD E8 00               883       [TOBUFF1]
                             884 ;
                             885 ;符号
                             886 ;
0532  3E 00 32 94 02         887       (符号1)=正
                             888 ;
                             889 ;指数/2
                             890 ;
0537  3A 95 02 B7 CA 72 04   891       A=(指数1)  IF A=0 JP [演算終了]
                             892 ;/2
053E  D6 81 1F               893       SUB A,BIAS  RRA
                             894 ;     PUSH AF
0541  38 06                  895       IF NC THEN
                             896 ;       CY=0
                             897         [>>] 仮数1
0543  21 96 02 CD 72 00      897+            HL=仮数1  CALL SHTR
                             898       FI
                             899 ;     POP AF
0549  C6 81 32 95 02         900       ADD BIAS  (指数1)=A
                             901 ;
                             902 ;仮数
                             903 ;
054E  21 96 02 11 A8 02 CD   904       HL=仮数1 DE=VAR.X  [LDIR]
0555  C0 02
0557  21 03 0A 11 96 02 CD   905       HL=値0   DE=仮数1  [LDIR]
055E  C0 02
0560  11 A0 02 CD C0 02      906                DE=仮数2  [LDIR]
0566  11 B8 02 CD C0 02      907                DE=平方根 [LDIR]
                             908 ;
056C  0E 21 3A 00 00 FE 08   909       C=8*4+1  IF (型)=倍精度 THEN C=8*7+1
0573  20 02 0E 39
                             910 ;
                             911       DO C {
                             912 ;       PUSH BC
                             913 ;
                             914         [<<0] 平方根
0577  21 BF 02 CD 8A 00      914+            HL=平方根+8-1  CALL RCF.SHTL
                             915 ;
                             916         [<<0] VAR.X
057D  21 AF 02 CD 8A 00      916+            HL=VAR.X+8-1  CALL RCF.SHTL
                             917         [<<]  仮数1
0583  21 9D 02 CD 8B 00      917+            HL=仮数1+8-1  CALL SHTL
                             918 ;
```

```
0589  21 AF 02 CD 8A 00      919+            HL=VAR.X+8-1  CALL RCF.SHTL
                             920         [<<]  仮数1
058F  21 9D 02 CD 8B 00      920+            HL=仮数1+8-1  CALL SHTL
                             921 ;
                             922         [<<1] 仮数2
0595  37 21 A7 02 CD 8B 00   922+            SCF HL=仮数2+8-1  CALL SHTL
                             923 ;
059C  CD 4B 02               924         [仮数比較]
                             925 ;
059F  30 10                  926         IF < THEN
05A1  21 A4 02               927           HL=仮数2+4
05A4  3A 00 00 FE 08 20 03   928           IF (型)=倍精度 THEN HL=仮数2+7
05AB  21 A7 02
                             929 ;
05AE  35                     930           DEC (HL) ;RES 0,(HL)
                             931 ;
05AF  18 21                  932         ELSE
05B1  CD C4 01               933           [仮数減算]
                             934 ;
05B4  3A 00 00 47            935           A=(型) B=A
                             936 ;
05B8  21 BC 02 11 A4 02      937           HL=平方根+4  DE=仮数2+4
05BE  FE 08 20 06            938           IF A=倍精度 THEN
05C2  21 BF 02 11 A7 02      939             HL=平方根+7  DE=仮数2+7
                             940           FI
05C8  CB FE                  941           SET 7,(HL)
                             942 ;         B=(型)
05CA  37 1A CE 00 12 1B 10   943           SCF  DO B {  ADC (DE),0  DEC DE }
05D1  F9
                             944         FI
                             945 ;
                             946 ;       POP BC
05D2  0D 20 A2               947       }
                             948 ;
05D5  21 B8 02 11 96 02 CD   949       HL=平方根  DE=仮数1  [LDIR]
05DC  C0 02
                             950 ;
05DE  C3 72 04               951       JP [演算終了]
                             952 ;
02B8:                        953 平方根  EQU VAR.Y
                             954 ;
                             955 ;****************************************
                             956 ;
                             957 ;     整数除算
                             958 ;
                             959 ;       [HL] = [HL] ¥ [DE]
                             960 ;
                             961 ;       保存 BC/DE/HL
                             962 ;
                             963 ;****************************************
                             964 ;
05E1:                        965 [IDIV]
                             966 ;
05E1  CD EB 03               967       [DIV]
                             968 ;     [FIX]
                             969 ;     RET
                             970 ;
                             971 ;****************************************
                             972 ;
                             973 ;     小数点以下切り捨て
                             974 ;
                             975 ;       [HL] = FIX [HL]
                             976 ;
                             977 ;       保存 BC/DE/HL
                             978 ;
                             979 ;****************************************
                             980 ;
05E4:                        981 [FIX]
                             982 ;
05E4  C5 D5 E5               983       PUSH BC/DE/HL
                             984 ;
                             985 ;指数チェック
                             986 ;
05E7  06 1F 11 04 00         987       B=8*4-1  DE=4
05EC  3A 00 00 FE 08 20 05   988       IF (型)=倍精度 THEN  B=8*7-1  DE=7
05F3  06 37 11 07 00
                             989 ;
05F8  7E D6 81 4F            990       A=(HL) SUB BIAS  C=A
                             991 ;
                             992 ;1未満
                             993 ;
05FC  30 09                  994       IF < THEN
05FE  11 03 0A EB CD C0 02   995         DE=値0  EX DE,HL [MOVE]
                             996 ;
                             997 ;1以上
                             998 ;
0605  18 1B                  999       ELSE
0607  B8 30 18              1000         IF A<B THEN
                            1001 ;         DE=4  IF (型)=倍精度 THEN  DE=7
060A  19                    1002           ADD HL,DE
                            1003 ;
060B  78 91                 1004           A=B SUB C
                            1005 ;A<>0
060D  FE 09 38 07           1006           UNTIL A<=8 {
0611  36 00 2B D6 08        1007             (HL)=0  DEC HL  SUB A,8
0616  18 F5                 1008           }
0618  0E FF CB 21 3D 20 FB  1009           C=$FF  DO A { SLA C }  AND (HL),C
061F  7E A1 77
                            1010 ;
                            1011 ;       ELSE ;小数を含まない
                            1012         FI
                            1013       FI
                            1014 ;
0622  E1 D1 C1              1015       POP HL/DE/BC
0625  C9                    1016       RET
                            1017 ;
                            1018 ;****************************************
                            1019 ;
                            1020 ;     引数の値の小数部
                            1021 ;
                            1022 ;       [HL] = FRAC [HL]
                            1023 ;
                            1024 ;       保存 BC/DE/HL
                            1025 ;
                            1026 ;****************************************
                            1027 ;
0626:                       1028 [FRAC]
                            1029 ;
0626  D5                    1030       PUSH DE
                            1031 ;
0627  11 37 06 CD C0 02     1032       DE=FRAC.X [MOVE]             ;小数点以下
062D  EB CD E4 05 EB        1033       EX DE,HL  [FIX]  EX DE,HL ;切り捨て
                            1034 ;
0632  CD DB 02              1035       [SUB]
                            1036 ;
0635  D1                    1037       POP DE
0636  C9                    1038       RET
                            1039 ;
0637: 00 00 00 00 00 00 00  1040 FRAC.X  DS BYTE
063E  00
                            1041 ;
                            1042 ;****************************************
                            1043 ;
                            1044 ;     小数点以下四捨五入
                            1045 ;
                            1046 ;       [HL] = CINT [HL]
                            1047 ;
                            1048 ;       保存 BC/DE/HL
                            1049 ;
                            1050 ;****************************************
                            1051 ;
063F:                       1052 [CINT]
                            1053 ;
063F  D5                    1054       PUSH DE
                            1055 ;
                            1056 ;小数部
                            1057 ;
0640  11 68 06 CD C0 02 EB  1058       DE=CINT.X [MOVE]  EX DE,HL
                            1059 ;     HL=CINT.X
0647  CD 26 06 7E FE 80     1060       [FRAC]  CP (HL),BIAS-1
064D  F5                    1061       PUSH AF
```

▶OS-9買いました。AVシェルを使っていると見たことのないファイル名が……。もしやと思い“chd /d1/.giftbox”とすると，うーん，福袋だったんですね。データディスクに隠れディレクトリなんてわかりませんよ，ふつう。　　　米村　謙一郎（25）長崎県

```
                           1062 ;
                           1063 ;小数点以下切り捨て
                           1064 ;
064E  21 37 06 CD C0 02 EB 1065       HL=FRAC.X  [MOVE]  EX DE.HL
                           1066 ;         ****
                           1067 ;
                           1068 ;四捨五入
                           1069 ;
0655  F1                   1070       POP AF
0656  D1                   1071       POP DE
                           1072 ;
                           1073 ;0.5未満
                           1074 ;
0657  C0                   1075       IF NZ RET
                           1076 ;
                           1077 ;0.5以上
                           1078 ;
0658  3A 94 02 FE 01 28 19 1079       IF (符号1)=負 JR [DEC]
                           1080 ;     IF (符号1)=負 : [DEC]
                           1081 ;     ELSE          : [INC]
                           1082 ;     FI
                           1083 ;     RET
                           1084 ;
                           1085 ;****************************************
                           1086 ;
                           1087 ;     [HL] = [HL] + 1
                           1088 ;
                           1089 ;     保存 BC/DE/HL
                           1090 ;
                           1091 ;****************************************
                           1092 ;
065F:                      1093 [INC]
                           1094 ;
065F  D5                   1095       PUSH DE
                           1096 ;
0660  11 FB 09 CD EA 02    1097       DE=値1 [ADD]
                           1098 ;
0666  D1                   1099       POP  DE
0667  C9                   1100       RET
                           1101 ;
0668: 00 00 00 00 00 00 00 1102 CINT.X  DS BYTE
066F  00
                           1103 ;
                           1104 ;****************************************
                           1105 ;
                           1106 ;     引数の値を越えない最大の整数
                           1107 ;
                           1108 ;       [HL] = INT [HL]
                           1109 ;
                           1110 ;       保存 BC/DE/HL
                           1111 ;
                           1112 ;****************************************
                           1113 ;
0670:                      1114 [INT]
                           1115 ;
0670  CD E4 05             1116       [FIX]
                           1117 ;
0673  23 CB 7E 2B C8       1118       INC HL  BIT 7,(HL)  DEC HL  IF Z RET
                           1119 ;
                           1120 ;負の場合
                           1121 ;
                           1122 ;     [DEC]
                           1123 ;     RET
                           1124 ;
                           1125 ;****************************************
                           1126 ;
                           1127 ;     [HL] = [HL] - 1
                           1128 ;
                           1129 ;     保存 BC/DE/HL
                           1130 ;
                           1131 ;****************************************
                           1132 ;
0678:                      1133 [DEC]
0678  D5                   1134       PUSH DE
                           1135 ;
0679  11 FB 09 CD DB 02    1136       DE=値1 [SUB]
                           1137 ;
067F  D1                   1138       POP  DE
0680  C9                   1139       RET
                           1140 ;
                           1141 ;****************************************
                           1142 ;
                           1143 ;     単精度 --> 倍精度
                           1144 ;
                           1145 ;       [HL]:単精度+3バイト
                           1146 ;
                           1147 ;        --> 倍精度 (単精度,0,0,0)
                           1148 ;
                           1149 ;       保存 BC/DE/HL
                           1150 ;
                           1151 ;****************************************
                           1152 ;
0681:                      1153 [CVDBL]
                           1154 ;
0681  E5                   1155       PUSH HL
                           1156 ;
0682  23 23 23 23 23       1157       INC HL/HL/HL/HL/HL
                           1158 ;
0687  AF                   1159       A=0
0688  77 23                1160       (HL)=A  INC HL
068A  77 23                1161       (HL)=A  INC HL
068C  77                   1162       (HL)=A
                           1163 ;
068D  E1                   1164       POP HL
068E  C9                   1165       RET
                           1166 ;
                           1167 ;****************************************
                           1168 ;
                           1169 ;     倍精度 --> 単精度
                           1170 ;
                           1171 ;       [HL]:倍精度 --> 単精度,0,0,0
                           1172 ;
                           1173 ;       保存 BC/DE/HL
                           1174 ;
                           1175 ;****************************************
                           1176 ;
068F:                      1177 [CVSNG]
                           1178 ;
068F  C5 D5 E5             1179       PUSH BC/DE/HL
                           1180 ;
0692  3A 00 00 F5          1181       A=(型)  PUSH AF
                           1182 ;
0696  3E 08 32 00 00 CD E8 1183       (型)=倍精度  [TOBUFF1]
069D  00
                           1184 ;
069E  3E 05 32 00 00 CD 23 1185       (型)=単精度  [正規化]
06A5  01
                           1186 ;
06A6  F1 32 00 00          1187       POP  AF  (型)=A
                           1188 ;
06AA  D1                   1189       POP  DE
06AB  D5                   1190       PUSH DE
                           1191 ;
06AC  21 95 02 01 05 00 ED 1192       HL=指数1  BC=5  LDIR
06B3  B0
                           1193 ;
06B4  AF                   1194       A=0
06B5  12 13                1195       (DE)=A  INC DE
06B7  12 13                1196       (DE)=A  INC DE
06B9  12                   1197       (DE)=A
                           1198 ;
06BA  E1 D1 C1             1199       POP HL/DE/BC
06BD  C9                   1200       RET
                           1201 ;
                           1202 ;****************************************
                           1203 ;
                           1204 ;     無符号整数 --> 浮動小数点
                           1205 ;
                           1206 ;     (0～65535)
                           1207 ;
                           1208 ;       [HL]:BUFF(5/8ﾊﾞｲﾄ) --> 浮動小数点
                           1209 ;       DE  :無符号整数値
                           1210 ;
                           1211 ;       保存 BC/DE/HL
                           1212 ;
                           1213 ;****************************************
                           1214 ;
06BE:                      1215 [CVUTF]
                           1216 ;
06BE  C5 D5 E5             1217       PUSH BC/DE/HL
                           1218 ;
06C1  D5                   1219       PUSH DE
                           1220 ;
06C2  21 03 0A CD E6 00    1221       HL=値0  [TOBUFF1]
                           1222 ;     (符号1)=正
06C8  3E 90 32 95 02       1223       (指数1)=BIAS+16-1
                           1224 ;
06CD  D1                   1225       POP  DE
06CE  6A                   1226       L=D
06CF  63                   1227       H=E
06D0  22 96 02             1228       (仮数1+0)=HL
                           1229 ;     (仮数1+0)=L=D
                           1230 ;     (仮数1+1)=H=E
                           1231 ;
06D3  C3 72 04             1232       JP [演算終了]
                           1233 ;
                           1234 ;****************************************
                           1235 ;
                           1236 ;     符号付整数 --> 浮動小数点
                           1237 ;
                           1238 ;     (-32768～32767)
                           1239 ;
                           1240 ;       [HL]:BUFF(5/8ﾊﾞｲﾄ) --> 浮動小数点
                           1241 ;       DE  :符号付整数値
                           1242 ;
                           1243 ;       保存 BC/DE/HL
                           1244 ;
                           1245 ;****************************************
                           1246 ;
06D6:                      1247 [CVITF]
                           1248 ;
06D6  CB 7A 28 E4          1249       IF BIT(7,D)=0 JR [CVUTF]
                           1250 ;
                           1251 ;負の場合
                           1252 ;
06DA  D5 CD 15 07 CD BE 06 1253       PUSH DE  [NEG.DE]  [CVUTF]  POP DE
06E1  D1
                           1254 ;     [NEG]
                           1255 ;     RET
                           1256 ;
                           1257 ;****************************************
                           1258 ;
                           1259 ;     符号反転
                           1260 ;
                           1261 ;       [HL] = -[HL]
                           1262 ;
                           1263 ;       保存 BC/DE/HL
                           1264 ;
                           1265 ;****************************************
                           1266 ;
06E2:                      1267 [NEG]
06E2  34 35 C8             1268       IF (HL)=0 RET
                           1269 ;
06E5  23 7E EE 80 77 2B    1270       INC HL  XOR (HL),$80  DEC HL
06EB  C9                   1271       RET
                           1272 ;
                           1273 ;****************************************
                           1274 ;
                           1275 ;     浮動小数点 --> 無符号整数
                           1276 ;
                           1277 ;     (0～65535)
                           1278 ;
                           1279 ;       [HL]:浮動小数点 --> DE:無符号整数値
                           1280 ;                           NZ:負
                           1281 ;                           Z :正
                           1282 ;       保存 BC/HL
                           1283 ;
                           1284 ;****************************************
                           1285 ;
06EC:                      1286 [CVFTU]
                           1287 ;
06EC  E5                   1288       PUSH HL
                           1289 ;
06ED  7E 23                1290       A=(HL)  INC HL ;指数
06EF  56 23                1291       D=(HL)  INC HL
06F1  5E 23                1292       E=(HL)  INC HL
06F3  66                   1293       H=(HL)
                           1294 ;
06F4  CB 7A                1295       BIT 7,D        ;符号
06F6  F5                   1296       PUSH AF
                           1297 ;
06F7  B7 28 14             1298       IF A<>0 THEN   ;指数
06FA  CB FA                1299         SET 7,D      ;ビット復活
                           1300 ;
06FC  FE 90 30 09          1301         UNTIL A>=BIAS+15 {
0700  CB 3A CB 1B CB 1C    1302           SRL D  RR E  RR H
0706  3C                   1303           INC A
0707  18 F3                1304         }
                           1305 ;小数点以下四捨五入
0709  CB 7C 28 01 13       1306         IF BIT(7,H)=1 THEN INC DE
                           1307       FI
                           1308 ;
070E  F1                   1309       POP AF ;負ならNZ
070F  E1                   1310       POP HL
0710  C9                   1311       RET
                           1312 ;
                           1313 ;****************************************
                           1314 ;
                           1315 ;     浮動小数点 --> 符号付整数
                           1316 ;
                           1317 ;     (-32768～32767)
                           1318 ;
                           1319 ;       [HL]:浮動小数点 --> DE:符号付整数値
                           1320 ;
                           1321 ;       保存 BC/HL
                           1322 ;
                           1323 ;****************************************
                           1324 ;
0711:                      1325 [CVFTI]
                           1326 ;
0711  CD EC 06 C8          1327       [CVFTU]  IF Z RET
                           1328 ;
                           1329 ;負の場合
                           1330 ;
                           1331 ;     [NEG.DE] ;DE=-DE
                           1332 ;     RET
                           1333 ;
0715:                      1334 [NEG.DE]
0715  7A 2F 57             1335       A=D CPL D=A
0718  7B 2F 5F 13 C9       1336       A=E CPL E=A  INC DE  RET
                           1337 ;
                           1338 ;****************************************
                           1339 ;
                           1340 ;     文字列 --> 浮動小数点
                           1341 ;
                           1342 ;       [HL]:BUFF(5/8ﾊﾞｲﾄ) --> 浮動小数点
                           1343 ;       [DE]:文字列        --> 次
                           1344 ;
                           1345 ;       保存 BC/HL
                           1346 ;
                           1347 ;****************************************
                           1348 ;
071D:                      1349 [CVSTF]
                           1350 ;
071D  C5                   1351       PUSH BC
                           1352 ;
071E  22 D2 07             1353       (STF.HL)=HL
                           1354 ;
                           1355 ;空白スキップ
                           1356 ;
0721  1A FE 20 20 03 13 18 1357       WHILE (DE)=" " { INC DE }
0728  F8
                           1358 ;
                           1359 ;符号
```

▶最近，レンタルCDの中に借りるものがなくなってきてしまった。ひどいときは，30分ほど探してとうとう1枚も借りずに帰ったこともあった。誰か，これはと思えるアーティストを教えてください。　　下川　貴弘（20）北海道

```
                           1360 ;
0729  CD C0 07 F5          1361      [STF符号] PUSH AF
                           1362 ;
                           1363 ;値=0
                           1364 ;
072D  D5                   1365      PUSH DE
072E  11 03 0A EB CD C0 02 1366      DE=値0  EX DE,HL [MOVE] EX DE,HL
0735  EB
0736  D1                   1367      POP  DE
                           1368 ;
                           1369 ;仮数部
                           1370 ;
0737  AF 32 D1 07          1371      (CNT.)=0
073B  0E 00                1372      C=0     ;ピリオドFLG
                           1373 ;     HL=(STF.HL)
                           1374      {
073D  0C 0D 20 07          1375        IF C=0 THEN
0741  1A FE 2E 20 02 13 0C 1376          IF (DE)="." THEN  INC DE/C ;C=1
                           1377        FI
                           1378 ;
0748  1A                   1379        A=(DE)
0749  FE 30 38 17          1380        IF A<"0" EXIT
074D  FE 3A 30 13          1381        IF A>"9" EXIT
                           1382 ;
                           1383 ;       HL=(STF.HL)
0751  CD 1A 05 1A D6 30 CD 1384        [MUL10]  A=(DE) SUB "0" [ADDA]
0758  00 05
                           1385 ;
075A  13                   1386        INC DE
                           1387 ;
075B  3A D1 07 91 32 D1 07 1388        SUB (CNT.),C ;IF C=1 THEN DEC (CNT.)
0762  18 D9                1389      }
                           1390 ;
                           1391 ;指数部
                           1392 ;
0764  1A                   1393      A=(DE)
                           1394 ;
0765  FE 45 28 04 FE 65 20 1395      IF A="E" OR A="e" THEN
076C  2E
076D  13                   1396        INC DE
                           1397 ;
076E  CD C0 07 F5          1398        [STF符号] PUSH AF
                           1399 ;
0772  21 00 00             1400        HL=0
                           1401        {
0775  1A D6 30             1402          A=(DE) SUB "0"
0778  38 11                1403          IF <   EXIT
077A  FE 0A 30 0D          1404          IF A>9 EXIT
                           1405 ;
077E  29 44 4D             1406          ADD HL,HL  BC=HL
0781  29                   1407          ADD HL,HL
0782  29                   1408          ADD HL,HL
0783  09                   1409          ADD HL,BC
0784  4F 06 00             1410          C=A B=0
0787  09                   1411          ADD HL,BC
                           1412 ;
0788  13                   1413          INC DE
0789  18 EA                1414        }
                           1415 ;
078B  F1 FE 01 20 04 7D ED 1416        POP AF  IF A=負 THEN  A=L NEG L=A
0792  44 6F
                           1417 ;
0794  3A D1 07 85 32 D1 07 1418        ADD (CNT.),L
                           1419      FI
                           1420 ;
                           1421 ;小数点処理
                           1422 ;
079B  2A D2 07             1423      HL=(STF.HL)
                           1424 ;
079E  3A D1 07             1425      A=(CNT.)
07A1  B7 28 14             1426      IF A<>0 THEN
07A4  FE 80 30 08 47 CD 1A 1427        IF A<$80 :      DO B,A { [MUL10] }
07AB  05 10 FB
07AE  18 08 ED 44 47 CD 23 1428        ELSE     : NEG  DO B,A { [DIV10] }
07B5  05 10 FB
                           1429        FI
                           1430      FI
                           1431 ;
                           1432 ;負の場合の処理
                           1433 ;
07B8  F1 FE 01 CC E2 06    1434      POP AF  IF A=負 [NEG]
                           1435 ;
07BE  C1                   1436      POP  BC
07BF  C9                   1437      RET
                           1438 ;
                           1439 ;
07C0:                      1440 [STF符号]
07C0  1A                   1441      A=(DE)
07C1  FE 2D 20 04 13 3E 01 1442      IF A="-" THEN  INC DE  A=負  RET
07C8  C9
                           1443 ;
07C9  FE 2B 20 01 13       1444      IF A="+" THEN  INC DE
07CE  3E 00                1445      A=正
07D0  C9                   1446      RET
                           1447 ;
07D1: 00                   1448 CNT.   DS 1
07D2: 00 00                1449 STF.HL DS 2
                           1450 ;
                           1451 ;****************************************
                           1452 ;
                           1453 ;     浮動小数点 --> 文字列
                           1454 ;
                           1455 ;       [HL]:浮動小数点
                           1456 ;       [DE]:BUFF         --> [DE]:文字列
                           1457 ;           (14/34バイト)
                           1458 ;
                           1459 ;       保存 BC/DE/HL
                           1460 ;
                           1461 ;****************************************
                           1462 ;
07D4:                      1463 [CVFTS]
                           1464 ;
07D4  C5 D5 E5             1465      PUSH BC/DE/HL
                           1466 ;
07D7  D5                   1467      PUSH DE
07D8  11 A8 02 CD C0 02 EB 1468      DE=VAR.X  [MOVE]  EX DE,HL
                           1469 ;     HL=VAR.X
07DF  D1                   1470      POP  DE
                           1471 ;
                           1472 ;ゼロ?
                           1473 ;
07E0  7E                   1474      A=(HL)
07E1  B7 20 0B             1475      IF A=0 THEN
07E4  3E 20 12 13          1476        (DE)=" " INC DE
07E8  3E 30 12 13 C3 17 09 1477        (DE)="0" INC DE  JP [CVFTS終了]
                           1478      FI
                           1479 ;
                           1480 ;符号
                           1481 ;
07EF  23 CB 7E CB BE       1482      INC HL  BIT 7,(HL)  RES 7,(HL)
                           1483 ;
07F4  3E 20 28 02 3E 2D    1484      A=" " IF NZ THEN A="-"
                           1485 ;
07FA  12 13                1486      (DE)=A  INC DE
                           1487 ;
07FC  D5                   1488      PUSH DE
                           1489 ;
                           1490 ;1 <= X < E8 (単精度) にする
                           1491 ;1 <= X < E16 (倍精度) にする
                           1492 ;
07FD  21 A8 02             1493      HL=VAR.X
                           1494 ;
0800  11 BB 09 0E 08       1495      DE=値E8  C=8
0805  3A 00 00 FE 08 20 05 1496      IF (型)=倍精度 THEN  DE=値E16  C=16
080C  11 7B 09 0E 10
                           1497 ;
0811  D5                   1498      PUSH DE
                           1499 ;
0812  06 00                1500      B=0 ;TEN指数
                           1501 ;1以上
0814  7E FE 81 38 0F       1502      IF (HL)>=BIAS+0 THEN
                           1503        {
0819  CD B6 04 38 08       1504          [CMP]  IF < EXIT
                           1505 ;
081E  CD EB 03 78 81 47    1506          [DIV]  ADD B,C
0824  18 F3                1507        }
                           1508 ;1未満
0826  18 0B                1509      ELSE
                           1510        {
0828  CD 5D 03 78 91 47    1511          [MUL]  SUB B,C
                           1512 ;
082E  7E FE 81 38 F5       1513        } UNTIL (HL)>=BIAS+0
                           1514      FI
0833  78 32 7A 09          1515      (TEN指数)=B
                           1516 ;
                           1517 ;E7  <= X < E8  (単精度) にする
                           1518 ;E15 <= X < E16 (倍精度) にする
                           1519 ;
0837  01 C3 09 3A 00 00 FE 1520      BC=値E7  IF (型)=倍精度 THEN BC=値E15
083E  08 20 03 01 83 09
0844  11 F3 09             1521      DE=値10
                           1522      {
0847  CD B6 04 38 17       1523        [CMP]  IF < EXIT
                           1524 ;
084C  3A 7A 09 3C 32 7A 09 1525        INC (TEN指数)
                           1526 ;
0853  7B D6 08 5F 30 01 15 1527        SUB E,BYTE IF C THEN DEC D ;SUB DE,BYTE
085A  79 C6 08 4F 30 01 04 1528        ADD C,BYTE IF C THEN INC B ;ADD BC,BYTE
0861  18 E4                1529      }
0863  50 59 CD 5D 03       1530      DE=BC [MUL]
                           1531 ;
                           1532 ;小数点以下4捨5入
                           1533 ;
0868  CD 3F 06             1534      [CINT]
                           1535 ;
086B  D1                   1536      POP DE
                           1537 ;     DE=値E8 IF (型)=倍精度 THEN DE=値E16
086C  CD B6 04             1538      [CMP]
086F  38 0A CD 23 05 3A 7A 1539      IF >= THEN  [DIV10]  INC (TEN指数)
0876  09 3C 32 7A 09
                           1540 ;
087B  D1                   1541      POP  DE
087C  3E 20 12 13          1542      (DE)=" " INC  DE
0880  D5                   1543      PUSH DE
                           1544 ;
                           1545 ;指数 -1～-8(-16)
                           1546 ;
0881  06 F8 3A 00 00 FE 08 1547      B=-8 IF (型)=倍精度 THEN B=-16
0888  20 02 06 F0
                           1548 ;
088C  3A 7A 09             1549      A=(TEN指数)
                           1550 ;
088F  B8 38 0A             1551      IF A>=B THEN
0892  ED 44                1552        NEG
0894  1B 47 3E 30 12 13 10 1553        DEC DE  DO B,A { (DE)="0"  INC DE }
089B  FA
                           1554      FI
                           1555 ;
                           1556 ;変換
                           1557 ;
089C  3A 00 00 FE 08 20 0C 1558      IF (型)=倍精度 THEN
08A3  01 9B 09 CD 1D 09    1559        BC=値E12 [FTS4桁]
08A9  01 BB 09 CD 1D 09    1560        BC=値E8  [FTS4桁]
                           1561      FI
                           1562 ;
08AF  01 DB 09 CD 1D 09    1563      BC=値E4  [FTS4桁]
08B5  21 A8 02 CD 44 09    1564      HL=VAR.X [FTS.PUT4]
                           1565 ;
                           1566 ;指数 0～7(15)
                           1567 ;
08BB  E1                   1568      POP HL
                           1569 ;
08BC  06 08 3A 00 00 FE 08 1570      B=8  IF (型)=倍精度 THEN  B=16
08C3  20 02 06 10
                           1571 ;
08C7  3A 7A 09 4F          1572      A=(TEN指数)  C=A
                           1573 ;
08CB  B8 30 0D             1574      IF A<B THEN
08CE  D5                   1575        PUSH DE
08CF  54 5D 1B             1576        DE=HL  DEC DE
                           1577 ;       C=A
08D2  06 00 03 ED B0 EB    1578        B=0    INC BC  LDIR  EX DE,HL
                           1579 ;       C=0
08D8  D1                   1580        POP  DE
                           1581 ;
                           1582 ;その他
                           1583 ;
08D9  18 0D                1584      ELSE
08DB  46 2B                1585        B=(HL) DEC HL
                           1586 ;
08DD  7E FE 30 20 04 0E 00 1587        IF (HL)="0" : C=0
08E4  18 02 70 23          1588        ELSE        : (HL)=B  INC HL
                           1589        FI
                           1590      FI
                           1591 ;
08E8  79 32 7A 09          1592      (TEN指数)=C
                           1593 ;
08EC  36 2E                1594      (HL)="."
                           1595 ;
                           1596 ;無駄な0や.をカット
                           1597 ;
08EE  1B 1A FE 30 28 FA    1598      { DEC DE  A=(DE) } UNTIL A<>"0"
                           1599 ;
08F4  FE 2E 28 01 13       1600      IF A<>"." THEN  INC DE
                           1601 ;
                           1602 ;指数
                           1603 ;
08F9  3A 7A 09 6F          1604      A=(TEN指数)  L=A
                           1605 ;
08FD  B7 28 17             1606      IF A<>0 THEN
0900  3E 45 12 13          1607        (DE)="E"  INC DE
0904  7D                   1608        A=L
0905  FE 80 30 04 3E 2B    1609        IF A<$80 :            A="+"
090B  18 05 ED 44 6F 3E 2D 1610        ELSE     : NEG L=A  A="-"
                           1611        FI
0912  12 13                1612        (DE)=A  INC DE
                           1613 ;       L=指数
0914  CD 66 09             1614        [FTS.PUT2]
                           1615      FI
                           1616 ;
0917:                      1617 [CVFTS終了]
0917  AF 12                1618      (DE)=0
                           1619 ;
0919  E1 D1 C1             1620      POP HL/DE/BC
091C  C9                   1621      RET
                           1622 ;
                           1623 ;
091D:                      1624 [FTS4桁]
                           1625 ;
091D  D5                   1626      PUSH DE
                           1627 ;
091E  21 A8 02 11 B8 02 CD 1628      HL=VAR.X  DE=VAR.Y [MOVE]  EX DE,HL
0925  C0 02 EB
                           1629 ;     HL=VAR.Y
0928  50 59 CD E1 05       1630      DE=BC      [IDIV]
092D  11 B0 02 CD C0 02 EB 1631      DE=VAR.XX [MOVE] EX DE,HL
                           1632 ;     HL=VAR.XX
0934  50 59 CD 5D 03 EB    1633      DE=BC      [MUL]  EX DE,HL
                           1634 ;     DE=VAR.XX
093A  21 A8 02 CD DB 02    1635      HL=VAR.X  [SUB]
                           1636 ;
0940  D1                   1637      POP DE
0941  21 B8 02             1638      HL=VAR.Y
                           1639 ;     [FTS.PUT4]
                           1640 ;     RET
                           1641 ;
0944:                      1642 [FTS.PUT4]
0944  D5 CD EC 06 EB D1    1643      PUSH DE  [CVFTU] EX DE,HL  POP DE
                           1644 ;
                           1645 ;HL:値
                           1646 ;
094A  01 E8 03 3E 2F B7    1647      BC=1000  A="0"-1  RCF
                           1648      {
0950  3C                   1649        INC A
```

▶最近やっとOh! XのことをOh! MZといい間違えなくなりました。こんな私が「昭和」から「平成」への変化に適応できるのはいつのことでしょうか？
本多　泰啓 (18) 大阪府

```
0951  ED 42                 1650          SBC HL,BC
0953  30 FB                 1651        } UNTIL C
0955  09 12 13              1652        ADD HL,BC  (DE)=A  INC DE
                            1653 ;
0958  01 64 00 3E 2F B7     1654        BC=100   A="0"-1  RCF
                            1655        {
095E  3C                    1656          INC A
095F  ED 42                 1657          SBC HL,BC
0961  30 FB                 1658        } UNTIL C
0963  09 12 13              1659        ADD HL,BC  (DE)=A  INC DE
                            1660 ;
                            1661 ;      [FTS.PUT2]
                            1662 ;      RET
                            1663 ;
0966:                       1664 [FTS.PUT2]
                            1665 ;
                            1666 ;L:値
                            1667 ;
0966  06 0A 7D 0E 2F        1668        B=10  A=L  C="0"-1
                            1669        {
096B  0C                    1670          INC C
096C  90                    1671          SUB A,B
096D  30 FC                 1672        } UNTIL C
096F  80 6F 79 12 13        1673        ADD A,B  L=A  (DE)=C  INC DE
                            1674 ;
0974  7D C6 30 12 13        1675        A=L ADD "0"   (DE)=A  INC DE
                            1676 ;
0979  C9                    1677        RET
                            1678 ;
097A: 00                    1679 TEN指数 DS 1
                            1680 ;
097B: B6 0E 1B C9 BF 04 00  1681 値E16 DB $B6,$0E,$1B,$C9,$BF,$04,$00,$00
0982  00
0983: B2 63 5F A9 31 A0 00  1682 値E15 DB $B2,$63,$5F,$A9,$31,$A0,$00,$00
098A  00
098B: AF 35 E6 20 F4 80 00  1683 値E14 DB $AF,$35,$E6,$20,$F4,$80,$00,$00
0992  00
0993: AC 11 84 E7 2A 00 00  1684 値E13 DB $AC,$11,$84,$E7.$2A,$00,$00,$00
099A  00
099B: A8 68 D4 A5 10 00 00  1685 値E12 DB $A8,$68,$D4,$A5,$10,$00,$00,$00
09A2  00
09A3: A5 3A 43 B7 40 00 00  1686 値E11 DB $A5,$3A,$43,$B7,$40,$00,$00,$00
09AA  00
09AB: A2 15 02 F9 00 00 00  1687 値E10 DB $A2,$15,$02,$F9,$00,$00,$00,$00
09B2  00
09B3: 9E 6E 6B 28 00 00 00  1688 値E9  DB $9E,$6E,$6B,$28,$00,$00,$00,$00
09BA  00
09BB: 9B 3E BC 20 00 00 00  1689 値E8  DB $9B,$3E,$BC,$20,$00,$00,$00,$00
09C2  00
09C3: 98 18 96 80 00 00 00  1690 値E7  DB $98,$18,$96,$80,$00,$00,$00,$00
09CA  00
09CB: 94 74 24 00 00 00 00  1691 値E6  DB $94,$74,$24,$00,$00,$00,$00,$00
09D2  00
09D3: 91 43 50 00 00 00 00  1692 値E5  DB $91,$43,$50,$00,$00,$00,$00,$00
09DA  00
09DB: 8E 1C 40 00 00 00 00  1693 値E4  DB $8E,$1C,$40,$00,$00,$00,$00,$00
09E2  00
09E3: 8A 7A 00 00 00 00 00  1694 値E3  DB $8A,$7A,$00,$00,$00,$00,$00,$00
09EA  00
09EB: 87 48 00 00 00 00 00  1695 値E2  DB $87,$48,$00,$00,$00,$00,$00,$00
09F2  00
09F3: 84 20 00 00 00 00 00  1696 値10  DB $84,$20,$00,$00,$00,$00,$00,$00
09FA  00
09FB: 81 00 00 00 00 00 00  1697 値1   DB $81,$00,$00,$00,$00,$00,$00,$00
0A02  00
0A03: 00 00 00 00 00 00 00  1698 値0   DB $00,$00,$00,$00,$00,$00,$00,$00
0A0A  00
                            1699 ;
                            1700 ;****************************************
                            1701 ;
                            1702 ;      TAN
                            1703 ;
                            1704 ;        [HL] = TAN [HL]
                            1705 ;
                            1706 ;        保存 BC/DE/HL
                            1707 ;
                            1708 ;****************************************
                            1709 ;
0A0B:                       1710 [TAN]
                            1711 ;
                            1712 ;TAN(X) = SIN(X)/COS(X)
                            1713 ;
0A0B  D5                    1714        PUSH DE
                            1715 ;
0A0C  11 1F 0A CD C0 02     1716        DE=TAN.X  [MOVE]
                            1717 ;
0A12  CD 32 0A EB CD 27 0A  1718        [SIN]  EX DE,HL [COS] EX DE,HL  [DIV]
0A19  EB CD EB 03
                            1719 ;
0A1D  D1                    1720        POP  DE
0A1E  C9                    1721        RET
                            1722 ;
0A1F: 00 00 00 00 00 00 00  1723 TAN.X  DS BYTE
0A26  00
                            1724 ;
                            1725 ;****************************************
                            1726 ;
                            1727 ;      COS
                            1728 ;
                            1729 ;        [HL] = COS [HL]
                            1730 ;
                            1731 ;        保存 BC/DE/HL
                            1732 ;
                            1733 ;****************************************
                            1734 ;
0A27:                       1735 [COS]
                            1736 ;
                            1737 ;COS(X) = SIN(X+pai/2)
                            1738 ;
0A27  C5 D5                 1739        PUSH BC/DE
                            1740 ;
0A29  E5                    1741        PUSH HL
                            1742 ;
0A2A  11 B9 0A CD EA 02 18  1743        DE=値PAI2 [ADD]  JR [SIN処理]
0A31  03
                            1744 ;
                            1745 ;****************************************
                            1746 ;
                            1747 ;      SIN
                            1748 ;
                            1749 ;        [HL] = SIN [HL]
                            1750 ;
                            1751 ;        保存 BC/DE/HL
                            1752 ;
                            1753 ;****************************************
                            1754 ;
0A32:                       1755 [SIN]
                            1756 ;
0A32  C5 D5                 1757        PUSH BC/DE
                            1758 ;
0A34  E5                    1759        PUSH HL
                            1760 ;
0A35:                       1761 [SIN処理]
                            1762 ;
                            1763 ;-2pai < X < 2pai にする
                            1764 ;
0A35  0E 00                 1765        C=0 ;符号反転フラグ
                            1766 ;
0A37  11 A8 02 CD C0 02 EB  1767        DE=VAR.X  [MOVE]  EX DE,HL
                            1768 ;      HL=VAR.X
0A3E  11 C9 0A CD 96 04     1769        DE=値2PAI [MOD]
                            1770 ;
                            1771 ;0 <= X < 2pai にする
                            1772 ;
0A44  23                    1773        INC HL
0A45  CB 7E 28 03 CB BE 0C  1774        IF BIT(7,(HL))=1 THEN RES 7,(HL) INC C
0A4C  2B                    1775        DEC HL
                            1776 ;
                            1777 ;0 <= X < pai にする
                            1778 ;
0A4D  11 C1 0A CD B6 04     1779        DE=値PAI  [CMP]
0A53  38 04 CD DB 02 0C     1780        IF >= THEN           [SUB]  INC C
                            1781 ;
                            1782 ;-pai/2 <= X < pai/2 にする
                            1783 ;
0A59  11 B9 0A CD B6 04     1784        DE=値PAI2 [CMP]
0A5F  38 07 11 C1 0A CD DB  1785        IF >= THEN  DE=値PAI [SUB]  INC C
0A66  02 0C
                            1786 ;
                            1787 ;符号反転
                            1788 ;
0A68  CB 41 C4 E2 06        1789        IF BIT(0,C)=1 [NEG]
                            1790 ;
                            1791 ;多項式より値を求める
                            1792 ;
                            1793 ;                 1  3  5  7
                            1794 ;                X  X  X  X
                            1795 ;      SIN(X) = ---+---+ …
                            1796 ;                1! 3! 5! 7!
                            1797 ;
                            1798 ;                  1   2  1   2 1   2 1
                            1799 ;            = X(-+X (--+X (-+X (-+ … ))))
                            1800 ;                  1!     3!    5!    7!
                            1801 ;
0A6D  7E FE 4D 30 0A        1802        IF (HL)<=$4C THEN
0A72  D1 21 03 0A CD C0 02  1803          POP DE  HL=値0  [MOVE]  EX DE,HL
0A79  EB
                            1804 ;
0A7A  18 3A                 1805        ELSE
                            1806 ;        HL=VAR.X
0A7C  11 B0 02 CD C0 02 EB  1807          DE=VAR.XX [MOVE] EX DE,HL [MUL]
0A83  CD 5D 03
                            1808 ;
0A86  D1 21 D1 0A CD C0 02  1809          POP DE  HL=SIN.TBL [MOVE] EX DE,HL
0A8D  EB
                            1810 ;        HL=引数
                            1811 ;        DE=SIN.TBL
0A8E  06 0A                 1812          B=10
0A90  3A 00 00 FE 05 20 05  1813          IF (型)=単精度 THEN
0A97  06 08 11 E1 0A        1814            B=8  DE=SIN.TBL+BYTE*2
                            1815          FI
                            1816 ;
                            1817          DO B {
0A9C  D5 11 B0 02 CD 5D 03  1818            PUSH DE  DE=VAR.XX [MUL]  POP DE
0AA3  D1
                            1819 ;
0AA4  7B C6 08 5F 30 01 14  1820            ADD E,BYTE IF C THEN INC D ;ADD DE,BYTE
0AAB  CD EA 02              1821            [ADD]
0AAE  10 EC                 1822          }
0AB0  11 A8 02 CD 5D 03     1823          DE=VAR.X  [MUL]
                            1824        FI
                            1825 ;
0AB6  D1 C1                 1826        POP DE/BC
0AB8  C9                    1827        RET
                            1828 ;
0AB9: 81 49 0F DA A2 21 68  1829 値PAI2 DB $81,$49,$0F,$DA,$A2,$21,$68,$C2
0AC0  C2
0AC1: 82 49 0F DA A2 21 68  1830 値PAI  DB $82,$49,$0F,$DA,$A2,$21,$68,$C2
0AC8  C2
0AC9: 83 49 0F DA A2 21 68  1831 値2PAI DB $83,$49,$0F,$DA,$A2,$21,$68,$C2
0AD0  C2
                            1832 ;
0AD1:                       1833 SIN.TBL
                            1834 ;      DB $1A,$12,$CF,$CC,$5A,$1A,$C5,$6B ; 1/29!
                            1835 ;      DB $23,$68,$D5,$8E,$16,$E6,$75,$19 ;-1/27!
                            1836 ;      DB $2D,$1F,$9E,$66,$E8,$B2,$FD,$46 ; 1/25!
                            1837 ;      DB $36,$BB,$0D,$A0,$98,$B1,$C0,$CF ;-1/23!
                            1838 ;10
0AD1  3F 38 DC 77 B6 E7 AB  1839       DB $3F,$38,$DC,$77,$B6,$E7,$AB,$8C ; 1/21!
0AD8  8C
0AD9  48 97 A4 DA 34 0A 0A  1840       DB $48,$97,$A4,$DA,$34,$0A,$0A,$B9 ;-1/19!
0AE0  B9
                            1841 ;8
0AE1  50 4A 96 3B 81 B5 6A  1842       DB $50,$4A,$96,$3B,$81,$85,$6A,$53 ; 1/17!
0AE8  53
0AE9  58 D7 3F 9F 39 9D C0  1843       DB $58,$D7,$3F,$9F,$39,$9D,$C0,$F9 ;-1/15!
0AF0  F9
0AF1  60 30 92 30 9D 43 68  1844       DB $60,$30,$92,$30,$9D,$43,$68,$4C ; 1/13!
0AF8  4C
0AF9  67 D7 32 2B 3F AA 27  1845       DB $67,$D7,$32,$2B,$3F,$AA,$27,$1C ;-1/11!
0B00  1C
0B01  6E 38 EF 1D 2A B6 39  1846       DB $6E,$38,$EF,$1D,$2A,$B6,$39,$9C ; 1/ 9!
0B08  9C
0B09  74 D0 0D 00 D0 0D 00  1847       DB $74,$D0,$0D,$00,$D0,$0D,$00,$D0 ;-1/ 7!
0B10  D0
0B11  7A 08 88 88 88 88 88  1848       DB $7A,$08,$88,$88,$88,$88,$88,$89 ; 1/ 5!
0B18  89
0B19  7E AA AA AA AA AA AA  1849       DB $7E,$AA,$AA,$AA,$AA,$AA,$AA,$AB ;-1/ 3!
0B20  AB
                            1850 ;
0B21  81 00 00 00 00 00 00  1851       DB $81,$00,$00,$00,$00,$00,$00,$00 ; 1/ 1!
0B28  00
                            1852 ;
                            1853 ;****************************************
                            1854 ;
                            1855 ;      ATN（アークタンジェント）
                            1856 ;
                            1857 ;        [HL] = ATN [HL]
                            1858 ;
                            1859 ;        保存 BC/DE/HL
                            1860 ;
                            1861 ;****************************************
                            1862 ;
0B29:                       1863 [ATN]
                            1864 ;
0B29  C5 D5                 1865        PUSH BC/DE
                            1866 ;
                            1867 ;符号を保存し、正の値にする
                            1868 ;
0B2B  23 CB 7E CB BE 2B     1869        INC HL  BIT 7,(HL)  RES 7,(HL)  DEC HL
0B31  F5                    1870        PUSH AF
                            1871 ;
                            1872 ;1より大きいか小さいかで処理を分ける
                            1873 ;
                            1874 ;      X<1 : ATN(X)
                            1875 ;      X=1 : pai/4
                            1876 ;      X>1 : -(ATN(1/X)-pai/2)
                            1877 ;
0B32  11 FB 09 CD B6 04     1878        DE=値1 [CMP]
                            1879 ;
0B38  30 05                 1880        IF < THEN
0B3A  CD 73 0B              1881          [ATN関数]
                            1882 ;
0B3D  18 2D 20 0B           1883        EF = THEN
0B41  11 B9 0A EB CD C0 02  1884          DE=値PAI2 EX DE,HL [MOVE] EX DE,HL
0B48  EB
0B49  35                    1885          DEC (HL)
                            1886 ;
0B4A  18 20                 1887        ELSE
0B4C  11 A8 02 CD C0 02     1888          DE=VAR.X  [MOVE]
0B52  11 FB 09 EB CD C0 02  1889          DE=値1    EX DE,HL [MOVE] EX DE,HL
0B59  EB
0B5A  11 A8 02 CD EB 03     1890          DE=VAR.X  [DIV]
0B60  CD 73 0B              1891          [ATN関数]
0B63  11 B9 0A CD DB 02     1892          DE=値PAI2 [SUB]
0B69  CD E2 06              1893          [NEG]
                            1894        FI
                            1895 ;
                            1896 ;引数が負の場合、負の値にする
                            1897 ;
0B6C  F1 C4 E2 06           1898        POP AF  IF NZ [NEG]
                            1899 ;
0B70  D1 C1                 1900        POP DE/BC
0B72  C9                    1901        RET
                            1902 ;
                            1903 ;
0B73:                       1904 [ATN関数]
                            1905 ;
                            1906 ;0.4142より大きい場合は
                            1907 ;
                            1908 ;ATN(0.4142)+ATN((X-0.4142)/(0.4142X+1))
                            1909 ;
0B73  11 E0 0B CD B6 04 F5  1910        DE=値.414  [CMP]  PUSH AF
                            1911 ;
0B7A  38 1B                 1912        IF >= THEN
0B7C  11 A8 02 CD C0 02     1913          DE=VAR.X  [MOVE]
```

▶スキーに出かけて身体はボロボロ。金は使い果たした。これではゲームが買えない。プリンタも買えない。資金がなくても遊べるようにゲームプログラムを載せて！

松浦　光宜（20）宮城県

```
                             1914 ;
0B82  11 E0 0B CD DB 02      1915         DE=値.414 [SUB]
0B88  E5                     1916         PUSH HL
0B89  21 A8 02 CD 5D 03 CD   1917         HL=VAR.X  [MUL] [INC]  EX DE,HL
0B90  5F 06 EB
0B93  E1 CD EB 03            1918         POP HL    [DIV]
                             1919       FI
                             1920 ;
                             1921 ;多項式より値を求める
                             1922 ;
                             1923 ;               1  3  5  7
                             1924 ;              X  X  X  X
                             1925 ;      ATN(X) = ―――+―――+ …
                             1926 ;              1  3  5  7
                             1927 ;
                             1928 ;                1   2  1   2 1   2  1
                             1929 ;             = X(―+X ̄(――+X (―+X (――+ … ))))
                             1930 ;                1      3      5      7
                             1931 ;X=引数
0B97  11 A8 02 CD C0 02      1932       DE=VAR.X   [MOVE]
                             1933 ;XX=X*X
0B9D  11 B0 02 CD C0 02 EB   1934       DE=VAR.XX  [MOVE]  EX DE,HL  [MUL]
0BA4  CD 5D 03
                             1935 ;
0BA7  21 F0 0B CD C0 02 EB   1936       HL=ATN.TBL [MOVE]  EX DE,HL
                             1937 ;     HL=引数
                             1938 ;     DE=ATN.TBL
0BAE  06 12                  1939       B=18
0BB0  3A 00 00 FE 05 20 05   1940       IF (型)=単精度 THEN
0BB7  06 0A 11 30 0C         1941         B=10  DE=ATN.TBL+BYTE*8
                             1942       FI
                             1943 ;
                             1944       DO B {
0BBC  D5 11 B0 02 CD 5D 03   1945         PUSH DE  DE=VAR.XX [MUL]  POP DE
0BC3  D1
                             1946 ;
0BC4  7B C6 08 5F 30 01 14   1947         ADD E,BYTE IF C THEN INC D ;ADD DE,BYTE
0BCB  CD EA 02               1948         [ADD]
0BCE  10 EC                  1949       }
0BD0  11 A8 02 CD 5D 03      1950       DE=VAR.X  [MUL]
                             1951 ;
                             1952 ;0.4142より大きい場合は + ATN(0.4142)
                             1953 ;
0BD6  F1                     1954       POP AF
0BD7  38 06 11 E8 0B CD EA   1955       IF >= THEN  DE=値ATN.414  [ADD]
0BDE  02
                             1956 ;
0BDF  C9                     1957       RET
                             1958 ;
0BE0: 7F 54 12 05 BC 01 A3   1959 値.414     DB $7F,$54,$12,$05,$BC,$01,$A3,$6E
0BE7  6E
0BE8: 7F 49 0E 56 32 C5 5B   1960 値ATN.414 DB $7F,$49,$0E,$56,$32,$C5,$5B,$5C
0BEF  5C
                             1961 ;
0BF0:                        1962 ATN.TBL
                             1963 ;     DB $7B,$36,$0B,$60,$B6,$0B,$60,$B6 ; 1/45
                             1964 ;     DB $7B,$BE,$B2,$FA,$0B,$E8,$2F,$A1 ;-1/43
                             1965 ;     DB $7B,$47,$CE,$0C,$7C,$E0,$C7,$CE ; 1/41
                             1966 ;     DB $7B,$D2,$0D,$20,$D2,$0D,$20,$D2 ;-1/39
                             1967 ;18
0BF0  7B 5D 67 C8 A6 0D D6   1968       DB $7B,$5D,$67,$C8,$A6,$0D,$D6,$7D ; 1/37
0BF7  7D
0BF8  7B EA 0E A0 EA 0E A0   1969       DB $7B,$EA,$0E,$A0,$EA,$0E,$A0,$EA ;-1/35
0BFF  EA
0C00  7B 78 3E 0F 83 E0 F8   1970       DB $7B,$78,$3E,$0F,$83,$E0,$F8,$3E ; 1/33
0C07  3E
0C08  7C 84 21 08 42 10 84   1971       DB $7C,$84,$21,$08,$42,$10,$84,$21 ;-1/31
0C0F  21
0C10  7C 0D 3D CB 08 D3 DC   1972       DB $7C,$0D,$3D,$CB,$08,$D3,$DC,$B1 ; 1/29
0C17  B1
0C18  7C 97 B4 25 ED 09 7B   1973       DB $7C,$97,$B4,$25,$ED,$09,$7B,$42 ;-1/27
0C1F  42
0C20  7C 23 D7 0A 3D 70 A3   1974       DB $7C,$23,$D7,$0A,$3D,$70,$A3,$D7 ; 1/25
0C27  D7
0C28  7C B2 16 42 C8 59 0B   1975       DB $7C,$B2,$16,$42,$C8,$59,$0B,$21 ;-1/23
0C2F  21
                             1976 ;10
0C30  7C 43 0C 30 C3 0C 30   1977       DB $7C,$43,$0C,$30,$C3,$0C,$30,$C3 ; 1/21
0C37  C3
0C38  7C D7 94 35 E5 0D 79   1978       DB $7C,$D7,$94,$35,$E5,$0D,$79,$43 ;-1/19
0C3F  43
0C40  7C 70 F0 F0 F0 F0 F0   1979       DB $7C,$70,$F0,$F0,$F0,$F0,$F0,$F1 ; 1/17
0C47  F1
0C48  7D 88 88 88 88 88 88   1980       DB $7D,$88,$88,$88,$88,$88,$88,$89 ;-1/15
0C4F  89
0C50  7D 1D 89 D8 9D 89 D8   1981       DB $7D,$1D,$89,$D8,$9D,$89,$D8,$9E ; 1/13
0C57  9E
0C58  7D BA 2E 8B A2 E8 BA   1982       DB $7D,$BA,$2E,$8B,$A2,$E8,$BA,$2F ;-1/11
0C5F  2F
0C60  7D 63 8E 38 E3 8E 38   1983       DB $7D,$63,$8E,$38,$E3,$8E,$38,$E4 ; 1/ 9
0C67  E4
0C68  7E 92 49 24 92 49 24   1984       DB $7E,$92,$49,$24,$92,$49,$24,$92 ;-1/ 7
0C6F  92
0C70  7E 4C CC CC CC CC CC   1985       DB $7E,$4C,$CC,$CC,$CC,$CC,$CC,$CD ; 1/ 5
0C77  CD
0C78  7F AA AA AA AA AA AA   1986       DB $7F,$AA,$AA,$AA,$AA,$AA,$AA,$AB ;-1/ 3
0C7F  AB
                             1987 ;
0C80  81 00 00 00 00 00 00   1988       DB $81,$00,$00,$00,$00,$00,$00,$00 ; 1/ 1
0C87  00
                             1989 ;
                             1990 ;****************************************
                             1991 ;
                             1992 ;       ＰＯＷ（べき乗）
                             1993 ;
                             1994 ;         [HL] = [HL] ^ [DE]
                             1995 ;
                             1996 ;         保存 BC/DE/HL
                             1997 ;
                             1998 ;****************************************
                             1999 ;
0C88:                        2000 [POW]
                             2001 ;
0C88  D5 E5                  2002       PUSH DE/HL
                             2003 ;
                             2004 ;第１引数を正にして、符号を保存
                             2005 ;
0C8A  23 CB 7E CB BE         2006       INC HL  BIT 7,(HL)  RES 7,(HL)
                             2007 ;
0C8F  3E 00 28 02 3E 01      2008       A=正  IF NZ THEN A=負
0C95  32 0E 0D               2009       (POW.符号)=A
                             2010 ;
                             2011 ;第２引数の小数部を求める
                             2012 ;
0C98  21 A8 02 EB CD C0 02   2013       HL=VAR.X  EX DE,HL [MOVE] EX DE,HL
0C9F  EB
                             2014 ;
0CA0  CD 26 06 7E            2015       [FRAC]  A=(HL) ;指数
                             2016 ;
                             2017 ;第２引数は小数を含むか、１６ビット以上
                             2018 ;
0CA4  E1 D1                  2019       POP HL/DE
                             2020 ;
0CA6  B7 20 05 1A FE 91 38   2021       IF A<>0 OR (DE)>BIAS+15 THEN
0CAD  08
                             2022 ;
0CAE  3A 0E 0D FE 01 C8      2023         IF (POW.符号)=負 RET ;演算不可
                             2024 ;
0CB4  18 59                  2025         JR [POW.exlog]
                             2026       FI
                             2027 ;
                             2028 ;第２引数は１６ビット以下の整数
                             2029 ;
0CB6  D5 C5                  2030       PUSH DE/BC
                             2031 ;
0CB8  E5                     2032       PUSH HL
0CB9  EB CD EC 06 42 4B EB   2033       EX DE,HL  [CVFTU] BC=DE  EX DE,HL
0CC0  E1                     2034       POP  HL
                             2035 ;
0CC1  C5                     2036       PUSH BC ;第２引数
                             2037 ;
                             2038 ;第２引数は５０以上
                             2039 ;
0CC2  79 D6 32 78 DE 00 38   2040       IF BC>=50 THEN
0CC9  05
0CCA  CD 0F 0D               2041         [POW.exlog]
                             2042 ;
                             2043 ;第２引数は５０以下
                             2044 ;
0CCD  18 2F                  2045       ELSE
0CCF  13                     2046         INC DE
0CD0  06 00 1A CB 7F 28 02   2047         B=正  IF BIT(7,(DE))=1 THEN  B=負
0CD7  06 01
                             2048 ;
0CD9  11 A8 02 CD C0 02      2049         DE=VAR.X  [MOVE]
                             2050 ;
0CDF  11 FB 09 EB CD C0 02   2051         DE=値1  EX DE,HL [MOVE] EX DE,HL
0CE6  EB
0CE7  11 A8 02               2052         DE=VAR.X
0CEA  0C 0D 28 10            2053         UNTIL C=0 {
0CEE  78 FE 00 20 05 CD 5D   2054           IF B=正 : [MUL]
0CF5  03
0CF6  18 03 CD EB 03         2055           ELSE    : [DIV]
                             2056           FI
0CFB  0D                     2057           DEC C
0CFC  18 EC                  2058         }
                             2059       FI
                             2060 ;
                             2061 ;第１引数が負で、第２引数が奇数の場合
                             2062 ;
0CFE  C1                     2063       POP BC ;第２引数
                             2064 ;
0CFF  3A 0E 0D FE 01 20 05   2065       IF (POW.符号)=負 THEN
                             2066 ;
0D06  CB 41 C4 E2 06         2067         IF BIT(0,C)=1 [NEG]
                             2068       FI
                             2069 ;
0D0B  C1 D1                  2070       POP BC/DE
0D0D  C9                     2071       RET
                             2072 ;
0D0E: 00                     2073 POW.符号 DS 1
                             2074 ;
                             2075 ;指数関数と対数関数を使って求める
                             2076 ;
0D0F:                        2077 [POW.exlog]
0D0F  CD F6 0D               2078       [LOG]
0D12  CD 5D 03               2079       [MUL]
                             2080 ;     [EXP]
                             2081 ;     RET
                             2082 ;
                             2083 ;****************************************
                             2084 ;
                             2085 ;       ＥＸＰ（指数関数）
                             2086 ;
                             2087 ;         [HL] = EXP [HL]
                             2088 ;
                             2089 ;         保存 BC/DE/HL
                             2090 ;
                             2091 ;****************************************
                             2092 ;
0D15:                        2093 [EXP]
                             2094 ;
0D15  C5 D5                  2095       PUSH BC/DE
                             2096 ;
                             2097 ;        x   x/LOGe2
                             2098 ;       e = 2
                             2099 ;
                             2100 ;X/LOGe2を求めて、整数部ｉと指数部ｆに分ける
                             2101 ;
                             2102 ;        x/LOGe2   i   f   i    f*LOGe2
                             2103 ;       2        = 2 * 2 = 2 * e
                             2104 ;
0D17  E5                     2105       PUSH HL
0D18  11 A8 02 CD C0 02 EB   2106       DE=VAR.X [MOVE] EX DE,HL
                             2107 ;     HL=VAR.X
0D1F  11 66 0D CD EB 03 CD   2108       DE=値Le2 [DIV]  [FRAC]
0D26  26 06
                             2109 ;
                             2110 ;  f*LOGe2
                             2111 ;e         を求める
                             2112 ;
0D28  11 66 0D CD 5D 03      2113       DE=値Le2 [MUL]
                             2114 ;
                             2115 ;多項式より値を求める
                             2116 ;
                             2117 ;               1  2  3  4
                             2118 ;        x     X  X  X  X
                             2119 ;       e  = 1+―+―+―+―+ …
                             2120 ;              1! 2! 3! 4!
                             2121 ;
                             2122 ;              1    1    1    1
                             2123 ;         = 1+X(―+X(―+X(―+X(― … ))))
                             2124 ;              1!   2!   3!   4!
                             2125 ;
0D2E  D1 21 6E 0D CD C0 02   2126       POP DE  HL=EXP.TBL [MOVE]  EX DE,HL
0D35  EB
                             2127 ;     HL=引数
                             2128 ;     DE=EXP.TBL
0D36  06 10                  2129       B=16
0D38  3A 00 00 FE 05 20 05   2130       IF (型)=単精度 THEN
0D3F  06 0C 11 8E 0D         2131         B=12  DE=EXP.TBL+BYTE*4
                             2132       FI
                             2133 ;
                             2134       DO B {
0D44  D5 11 A8 02 CD 5D 03   2135         PUSH DE  DE=VAR.X [MUL]  POP DE
0D4B  D1
                             2136 ;
0D4C  7B C6 08 5F 30 01 14   2137         ADD E,BYTE IF C THEN INC D ;ADD DE,BYTE
0D53  CD EA 02               2138         [ADD]
0D56  10 EC                  2139       }
                             2140 ;  i
                             2141 ;*2    ;整数部intを指数に加える
                             2142 ;
0D58  E5                     2143       PUSH HL
0D59  21 37 06 CD 11 07      2144       HL=FRAC.X  [CVFTI]
                             2145 ;        ****
0D5F  E1 7E 83 77            2146       POP  HL  ADD (HL),E
                             2147 ;
0D63  D1 C1                  2148       POP DE/BC
0D65  C9                     2149       RET
                             2150 ;
0D66: 80 31 72 17 F7 D1 CF   2151 値Le2 DB $80,$31,$72,$17,$F7,$D1,$CF,$7A
0D6D  7A
                             2152 ;
0D6E:                        2153 EXP.TBL
                             2154 ;     DB $43,$72,$A1,$5D,$20,$10,$11,$28 ;1/20!
                             2155 ;     DB $48,$17,$A4,$DA,$34,$0A,$0A,$B9 ;1/19!
                             2156 ;     DB $4C,$34,$13,$C3,$1D,$CB,$EC,$BC ;1/18!
                             2157 ;     DB $50,$4A,$96,$3B,$81,$85,$6A,$53 ;1/17!
                             2158 ;16
0D6E  54 57 3F 9F 39 9D C0   2159       DB $54,$57,$3F,$9F,$39,$9D,$C0,$F9 ;1/16!
0D75  F9
0D76  58 57 3F 9F 39 9D C0   2160       DB $58,$57,$3F,$9F,$39,$9D,$C0,$F9 ;1/15!
0D7D  F9
0D7E  5C 49 CB A5 46 03 E4   2161       DB $5C,$49,$CB,$A5,$46,$03,$E4,$E9 ;1/14!
0D85  E9
0D86  60 30 92 30 9D 43 68   2162       DB $60,$30,$92,$30,$9D,$43,$68,$4C ;1/13!
0D8D  4C
                             2163 ;12
0D8E  64 0F 76 C7 7F C6 C4   2164       DB $64,$0F,$76,$C7,$7F,$C6,$C4,$BE ;1/12!
0D95  BE
0D96  67 57 32 2B 3F AA 27   2165       DB $67,$57,$32,$2B,$3F,$AA,$27,$1C ;1/11!
0D9D  1C
0D9E  6B 13 F2 7D BB C4 FA   2166       DB $6B,$13,$F2,$7D,$BB,$C4,$FA,$E4 ;1/10!
0DA5  E4
0DA6  6E 38 EF 1D 2A B6 39   2167       DB $6E,$38,$EF,$1D,$2A,$B6,$39,$9C ;1/ 9!
0DAD  9C
0DAE  71 50 0D 00 D0 0D 00   2168       DB $71,$50,$0D,$00,$D0,$0D,$00,$D0 ;1/ 8!
0DB5  D0
0DB6  74 50 0D 00 D0 0D 00   2169       DB $74,$50,$0D,$00,$D0,$0D,$00,$D0 ;1/ 7!
0DBD  D0
0DBE  77 36 0B 60 B6 0B 60   2170       DB $77,$36,$0B,$60,$B6,$0B,$60,$B6 ;1/ 6!
0DC5  B6
0DC6  7A 08 88 88 88 88 88   2171       DB $7A,$08,$88,$88,$88,$88,$88,$89 ;1/ 5!
0DCD  89
0DCE  7C 2A AA AA AA AA AA   2172       DB $7C,$2A,$AA,$AA,$AA,$AA,$AA,$AB ;1/ 4!
```

THE SENTINEL

▶ZEDAを持っていなかったので２月号のREDAを入力中である。
小川　哲也（36）兵庫県

```
0DD5  AB
0DD6  7E 2A AA AA AA AA AA  2173       DB $7E,$2A,$AA,$AA,$AA,$AA,$AA,$AB ;1/ 3!
0DDD  AB
0DDE  80 00 00 00 00 00 00  2174       DB $80,$00,$00,$00,$00,$00,$00,$00 ;1/ 2!
0DE5  00
0DE6  81 00 00 00 00 00 00  2175       DB $81,$00,$00,$00,$00,$00,$00,$00 ;1/ 1!
0DED  00
                            2176 ;
0DEE  81 00 00 00 00 00 00  2177       DB $81,$00,$00,$00,$00,$00,$00,$00 ;1
0DF5  00
                            2178 ;
                            2179 ;****************************************
                            2180 ;
                            2181 ;      LOG（対数関数）
                            2182 ;
                            2183 ;        [HL] = LOG [HL]
                            2184 ;
                            2185 ;        保存 BC/DE/HL
                            2186 ;
                            2187 ;****************************************
                            2188 ;
0DF6:                       2189 [LOG]
                            2190 ;
0DF6  C5 D5                 2191       PUSH BC/DE
                            2192 ;
                            2193 ;Xを仮数部Mと指数部Sに分けて
                            2194 ;                             S
                            2195 ;     LOG e X = LOG e ( M * 2 )
                            2196 ;
                            2197 ;             = LOG e M + S * LOG e 2
                            2198 ;
0DF8  E5                    2199       PUSH HL
                            2200 ;
0DF9  11 A8 02 CD C0 02 EB  2201       DE=VAR.X [MOVE] EX DE,HL
                            2202 ;     HL=VAR.I
0E00  7E 32 70 0E 36 81     2203       (LOG指数)=(HL)  (HL)=BIAS+0
                            2204 ;
                            2205 ;LOG e M を多項式より求める
                            2206 ;
                            2207 ;        M-1
                            2208 ;     h = ―
                            2209 ;        M+1
                            2210 ;               1  3  5  7
                            2211 ;              h  h  h  h
                            2212 ;     LOG e M = 2(―+―+―+―+ … )
                            2213 ;               1  3  5  7
                            2214 ;
                            2215 ;                 1  2 1  2 1  2 1
                            2216 ;             = 2(h(―+h (―+h (―+h (―+ … )))))
                            2217 ;                 1    3    5    7
                            2218 ;     HL=VAR.X
0E06  11 B8 02 CD C0 02     2219       DE=VAR.Y [MOVE]
                            2220 ;
0E0C  CD 78 06 EB CD 5F 06  2221       [DEC]  EX DE,HL [INC] EX DE,HL  [DIV]
0E13  EB CD EB 03
                            2222 ;XX=X*X
0E17  11 B0 02 CD C0 02 EB  2223       DE=VAR.XX [MOVE]  EX DE,HL  [MUL]
0E1E  CD 5D 03
                            2224 ;
0E21  D1 21 71 0E CD C0 02  2225       POP DE  HL=LOG.TBL [MOVE]  EX DE,HL
0E28  EB
                            2226 ;     HL=引数
                            2227 ;     DE=LOG.TBL
0E29  06 0E                 2228       B=14
0E2B  3A 00 00 FE 05 20 05  2229       IF (型)=単精度 THEN
0E32  06 08 11 A1 0E        2230         B=8   DE=LOG.TBL+BYTE*6
                            2231       FI
                            2232 ;
                            2233       DO B {
0E37  D5 11 B0 02 CD 5D 03  2234         PUSH DE  DE=VAR.XX [MUL]  POP DE
0E3E  D1
                            2235 ;
0E3F  7B C6 08 5F 30 01 14  2236         ADD E,BYTE IF C THEN INC D ;ADD DE,BYTE
0E46  CD EA 02              2237         [ADD]
0E49  10 EC                 2238       }
0E4B  11 A8 02 CD 5D 03     2239       DE=VAR.X [MUL]
                            2240 ;*2
0E51  34                    2241       INC (HL)
                            2242 ;
                            2243 ;+S*LOGe2
                            2244 ;
0E52  E5                    2245       PUSH HL
                            2246 ;
0E53  3A 70 0E D6 81        2247       A=(LOG指数) SUB BIAS
0E58  5F                    2248       E=A
0E59  7A 9A 57              2249       SBC D,D ;D=0 IF C THEN D=$FF
0E5C  21 B8 02 CD D6 06     2250       HL=VAR.Y [CVITF]
0E62  11 66 0D CD 5D 03     2251       DE=値Le2 [MUL]
0E68  EB                    2252       EX DE,HL
                            2253 ;
0E69  E1 CD EA 02           2254       POP HL  [ADD]
                            2255 ;
0E6D  D1 C1                 2256       POP DE/BC
0E6F  C9                    2257       RET
                            2258 ;
0E70: 00                    2259 LOG指数 DS 1
                            2260 ;
0E71:                       2261 LOG.TBL
                            2262 ;     DB $7B,$5D,$67,$C8,$A6,$0D,$D6,$7D ;1/37
                            2263 ;     DB $7B,$6A,$0E,$A0,$EA,$0E,$A0,$EA ;1/35
                            2264 ;     DB $7B,$78,$3E,$0F,$83,$E0,$F8,$3E ;1/33
                            2265 ;     DB $7C,$04,$21,$08,$42,$10,$84,$21 ;1/31
                            2266 ;14
0E71  7C 0D 3D CB 08 D3 DC  2267       DB $7C,$0D,$3D,$CB,$08,$D3,$DC,$B1 ;1/29
0E78  B1
0E79  7C 17 B4 25 ED 09 7B  2268       DB $7C,$17,$B4,$25,$ED,$09,$7B,$42 ;1/27
0E80  42
0E81  7C 23 D7 0A 3D 70 A3  2269       DB $7C,$23,$D7,$0A,$3D,$70,$A3,$D7 ;1/25
0E88  D7
0E89  7C 32 16 42 C8 59 0B  2270       DB $7C,$32,$16,$42,$C8,$59,$0B,$21 ;1/23
0E90  21
0E91  7C 43 0C 30 C3 0C 30  2271       DB $7C,$43,$0C,$30,$C3,$0C,$30,$C3 ;1/21
0E98  C3
0E99  7C 57 94 35 E5 0D 79  2272       DB $7C,$57,$94,$35,$E5,$0D,$79,$43 ;1/19
0EA0  43
                            2273 ;8
0EA1  7C 70 F0 F0 F0 F0 F0  2274       DB $7C,$70,$F0,$F0,$F0,$F0,$F0,$F1 ;1/17
0EA8  F1
0EA9  7D 08 88 88 88 88 88  2275       DB $7D,$08,$88,$88,$88,$88,$88,$89 ;1/15
0EB0  89
0EB1  7D 1D 89 D8 9D 89 D8  2276       DB $7D,$1D,$89,$D8,$9D,$89,$D8,$9E ;1/13
0EB8  9E
0EB9  7D 3A 2E 8B A2 E8 BA  2277       DB $7D,$3A,$2E,$8B,$A2,$E8,$BA,$2F ;1/11
0EC0  2F
0EC1  7D 63 8E 38 E3 8E 38  2278       DB $7D,$63,$8E,$38,$E3,$8E,$38,$E4 ;1/ 9
0EC8  E4
0EC9  7E 12 49 24 92 49 24  2279       DB $7E,$12,$49,$24,$92,$49,$24,$92 ;1/ 7
0ED0  92
0ED1  7E 4C CC CC CC CC CC  2280       DB $7E,$4C,$CC,$CC,$CC,$CC,$CC,$CD ;1/ 5
0ED8  CD
0ED9  7F 2A AA AA AA AA AA  2281       DB $7F,$2A,$AA,$AA,$AA,$AA,$AA,$AB ;1/ 3
0EE0  AB
                            2282 ;
0EE1  81 00 00 00 00 00 00  2283       DB $81,$00,$00,$00,$00,$00,$00,$00 ;1/ 1
0EE8  00
                            2284 ;
                            2285 ;****************************************
                            2286 ;
                            2287 ;     絶対値
                            2288 ;
                            2289 ;       [HL] = ABS [HL]
                            2290 ;
                            2291 ;       保存 BC/DE/HL
                            2292 ;
                            2293 ;****************************************
                            2294 ;
0EE9:                       2295 [ABS]
                            2296 ;
0EE9  23 CB BE 2B C9        2297       INC HL  RES 7,(HL)  DEC HL  RET
                            2298 ;
                            2299 ;****************************************
                            2300 ;
                            2301 ;     符号を返す
                            2302 ;
                            2303 ;       [HL] = SGN [HL]
                            2304 ;
                            2305 ;       [HL] > 0  -->  [HL] = 1
                            2306 ;       [HL] = 0  -->  [HL] = 0
                            2307 ;       [HL] < 0  -->  [HL] = -1
                            2308 ;
                            2309 ;       保存 BC/DE/HL
                            2310 ;
                            2311 ;****************************************
                            2312 ;
0EEE:                       2313 [SGN]
                            2314 ;
0EEE  34 35 C8              2315       IF (HL)=0 RET
                            2316 ;
0EF1  D5                    2317       PUSH DE
                            2318 ;
0EF2  23 CB 7E 2B           2319       INC HL  BIT 7,(HL)  DEC HL  ;符号
                            2320 ;
0EF6  F5                    2321       PUSH AF
0EF7  11 FB 09 EB CD C0 02  2322       DE=値1  EX DE,HL  [MOVE]  EX DE,HL
0EFE  EB
0EFF  F1                    2323       POP  AF
                            2324 ;
0F00  C4 E2 06              2325       IF NZ [NEG] ;負なら -1
                            2326 ;
0F03  D1                    2327       POP DE
0F04  C9                    2328       RET
                            2329 ;
                            2330 ;****************************************
                            2331 ;
                            2332 ;     円周率の引数倍
                            2333 ;
                            2334 ;       [HL] = PAI [HL]
                            2335 ;
                            2336 ;       保存 BC/DE/HL
                            2337 ;
                            2338 ;****************************************
                            2339 ;
0F05:                       2340 [PAI]
                            2341 ;
                            2342 ;PAI(X) = X*pai
                            2343 ;
0F05  D5 18 07              2344       PUSH DE  JR [RAD.1]
                            2345 ;
                            2346 ;****************************************
                            2347 ;
                            2348 ;     度 --> ラジアン
                            2349 ;
                            2350 ;       [HL] = RAD [HL]
                            2351 ;
                            2352 ;       保存 BC/DE/HL
                            2353 ;
                            2354 ;****************************************
                            2355 ;
0F08:                       2356 [RAD]
                            2357 ;
                            2358 ;RAD(X) = X*pai/180
                            2359 ;
0F08  D5                    2360       PUSH DE
                            2361 ;
0F09  11 17 0F CD EB 03     2362       DE=値180 [DIV]
                            2363 ;
0F0F:                       2364 [RAD.1]
0F0F  11 C1 0A CD 5D 03     2365       DE=値PAI [MUL]
                            2366 ;
0F15  D1                    2367       POP DE
0F16  C9                    2368       RET
                            2369 ;
0F17: 88 34 00 00 00 00 00  2370 値180 DB $88,$34,$00,$00,$00,$00,$00,$00
0F1E  00
                            2371 ;
                            2372 ;****************************************
                            2373 ;
                            2374 ;これ以降は、ＢＵＧパッチ用エリアです。
                            2375 ;
                            2376 ;何やら意味のないＤＷ命令が並んでいます
                            2377 ;
                            2378 ;が、これはリロケータブルオブジェクトに
                            2379 ;
                            2380 ;パッチを当てるためのオマジナイです。あ
                            2381 ;
                            2382 ;まり気にしないでください。
                            2383 ;
                            2384       REPT  0+(ｱﾄﾞﾚｽ+$1000-$)/6
                            2385         DW $,$,$
                            2386       ENDM
0F1F  1F 0F 1F 0F 1F 0F     2386+       DW $,$,$
0F25  25 0F 25 0F 25 0F     2386+       DW $,$,$
0F2B  2B 0F 2B 0F 2B 0F     2386+       DW $,$,$
0F31  31 0F 31 0F 31 0F     2386+       DW $,$,$
0F37  37 0F 37 0F 37 0F     2386+       DW $,$,$
0F3D  3D 0F 3D 0F 3D 0F     2386+       DW $,$,$
0F43  43 0F 43 0F 43 0F     2386+       DW $,$,$
0F49  49 0F 49 0F 49 0F     2386+       DW $,$,$
0F4F  4F 0F 4F 0F 4F 0F     2386+       DW $,$,$
0F55  55 0F 55 0F 55 0F     2386+       DW $,$,$
0F5B  5B 0F 5B 0F 5B 0F     2386+       DW $,$,$
0F61  61 0F 61 0F 61 0F     2386+       DW $,$,$
0F67  67 0F 67 0F 67 0F     2386+       DW $,$,$
0F6D  6D 0F 6D 0F 6D 0F     2386+       DW $,$,$
0F73  73 0F 73 0F 73 0F     2386+       DW $,$,$
0F79  79 0F 79 0F 79 0F     2386+       DW $,$,$
0F7F  7F 0F 7F 0F 7F 0F     2386+       DW $,$,$
0F85  85 0F 85 0F 85 0F     2386+       DW $,$,$
0F8B  8B 0F 8B 0F 8B 0F     2386+       DW $,$,$
0F91  91 0F 91 0F 91 0F     2386+       DW $,$,$
0F97  97 0F 97 0F 97 0F     2386+       DW $,$,$
0F9D  9D 0F 9D 0F 9D 0F     2386+       DW $,$,$
0FA3  A3 0F A3 0F A3 0F     2386+       DW $,$,$
0FA9  A9 0F A9 0F A9 0F     2386+       DW $,$,$
0FAF  AF 0F AF 0F AF 0F     2386+       DW $,$,$
0FB5  B5 0F B5 0F B5 0F     2386+       DW $,$,$
0FBB  BB 0F BB 0F BB 0F     2386+       DW $,$,$
0FC1  C1 0F C1 0F C1 0F     2386+       DW $,$,$
0FC7  C7 0F C7 0F C7 0F     2386+       DW $,$,$
0FCD  CD 0F CD 0F CD 0F     2386+       DW $,$,$
0FD3  D3 0F D3 0F D3 0F     2386+       DW $,$,$
0FD9  D9 0F D9 0F D9 0F     2386+       DW $,$,$
0FDF  DF 0F DF 0F DF 0F     2386+       DW $,$,$
0FE5  E5 0F E5 0F E5 0F     2386+       DW $,$,$
0FEB  EB 0F EB 0F EB 0F     2386+       DW $,$,$
0FF1  F1 0F F1 0F F1 0F     2386+       DW $,$,$
0FF7  F7 0F F7 0F F7 0F     2386+       DW $,$,$
                            2387 ;
                            2388 $IF $<>ｱﾄﾞﾚｽ+$1000
                            2389 ;
0FFD  00 00 00              2390       DS ｱﾄﾞﾚｽ+$1000-$
                            2391 ;
                            2392 $ENDIF
                            2393 ;
                            2394 ;このプログラムは、
                            2395 ;
                            2396 ;幻と言われた
                            2397 ;
                            2398 ;（要するに、投稿したがボツになった）
                            2399 ;
                            2400 ;自称最強のアセンブラ
                            2401 ;
                            2402 ;（要するに、誰も認めてくれない）
                            2403 ;
                            2404 ;「ＯＨＭ－Ｚ８０」で記述しました
                            2405 ;
                            2406 ;（要するに、私は未練がましい）。
                            2407 ;
                            2408       END

ﾊｯｼｭ TBL: 0116
ｼﾝﾎﾞﾙTBL: 0608
ｽﾄﾗｸﾄTBL: 01E0
ﾏｸﾛ  TBL: 00C9
ﾏｸﾛ  ﾜｰｸ: 0013

ｵﾌﾞｼﾞｪｸﾄ: 0000 [8000] - 0FFF [8FFF]
```

MZ-2500用 スペースハリアー・メインテーマ ©SEGA

Tajima Naoto
田島 直人

X1/X1turbo用 ANGEL

Sasaki Koji
佐々木 孝司

X68000用 Moonlight Feels Right

Ando Masahiro
安藤 正洋

X68000用 消えた王様の杖(ソーサリアン)〈生還〉 ©日本ファルコム

Kitano Naoyuki
北野 直之

力作を期待するといってるうち，あっという間にすごい作品ばかりになってしまって選ぶのに一苦労です。欲しいと思った音をピシャリと出すのはなかなか難しいと思うけど，皆さんがんばっていらっしゃいますね。いろいろな曲を楽しんで，いろいろな音を試してみてください。

おなじみのスペハリ

スペースハリアーのメインテーマは，これまでにもいろいろなところでミュージックプログラムとして紹介されてきました。今月はMZ-2500にも登場。ベースやスネアドラムの音がなかなかに気分を盛り上げてくれます。

プログラムの実行にはMMLの拡張を行い，好きな音色を選んで聞いてください。

作者の田島君は「高校受験の直前なのにLIVE in '89を盛り上げようとがんばって」くれたそうです。どうもありがとう。でも勉強のほうもがんばってください。

ポピュラーファンの皆さんへ

MusicBASICにはポピュラーソングから。もとBOØWYのメンバー氷室京介のANGELです。

原曲はかなり威勢のいいボーカルのロックンロールでした。こちらはインストゥルメンタル，FM音源8声と，PSG1声にハイハットを当てた構成。雰囲気のいいBGMとしても楽しめます。

プログラムの実行には，リスト2 ANGEL[A]とリスト3 ANGEL[B]が必要になります。2つを同じディスクにセーブしたあと,ANGEL[A]をrunして聞いてください。

MusicBASICにもたくさん作品が届き始めました。やっぱり自分で作るのって楽しいでしょ。

X-BASICには2曲

まずは，すでにスタンダードナンバーともいえるポピュラーの名曲Moonlight Feels Right。オリジナルは1979年にスターバックというグループがヒットさせたそうで，今回のプログラムは高橋幸宏が歌うバージョンから。

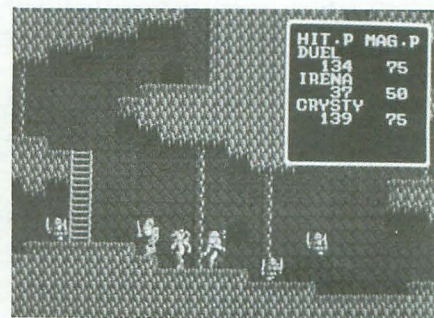
この曲のポイントはマリンバのリズミカルな演奏ですが，プログラムでもこれがとても聞きごたえのある仕上がりとなっています。シンプルなトランペットにしてみた，というリード音もよくマッチしていますね。

マリンバは「チャンネルが足りなくてステレオにできなかったのが残念」とのこと。でも，音色にはシロフォンの雰囲気もあって力の入っているのがわかります。

さてX-BASICの2曲目は，ゲームミュージックとしての登場頻度でも1，2位を争うソーサリアンから"消えた王様の杖〈生還〉"です。

作者の北野君は中学3年生。曲を聞いて簡単に音を拾ってしまうという頼もしい読者です。今回のソーサリアンでは，FM音源を駆使した細かい音色の設定に感心しました。リスト中にいろいろコメントを書いてくれているので，皆さんも参考にしてください。

ここ2カ月ほどはゲームとポピュラーで占められたLIVEですが，クラシックのほうもまた期待していますよ。

スペースハリアー

ソーサリアン

リスト1 SPACE HARRIER

```
1000 '
1010 ' SPACE  HARRIER  "Theme          (C)SEGA  1985
1020 '
1030 '
1040 '   初期設定・音色定義
1050 '
1060 if peek(&HFFF)-0 then print "PLAY文が拡張されていません。":end
1070 play init:clear max:dim A%(4,9)
1080 ST-peek@(0,&HFFF)+1:AD-0
1090 for K-0 to 6
1100 for I-0 to 4:for J-0 to 9
1110 read A%(I,J):next J,I
1120 for J-0 to 9:swap A%(2,J),A%(3,J):next
1130 for I-1 to 4:poke@ ST,AD,A%(I,5):AD-AD+1:next
1140 for I-1 to 4:poke@ ST,AD,A%(I,7)+(A%(I,8) and 7)*$10:AD-AD
+1:next
 1150 for I-1 to 4:poke@ ST,AD,A%(I,0)+A%(I,6)*$40:AD-AD+1:next
 1160 for I-1 to 4:poke@ ST,AD,A%(I,1)+A%(I,9)*$40:AD-AD+1:next
 1170 for I-1 to 4:poke@ ST,AD,A%(I,2):AD-AD+1:next
 1180 for I-1 to 4:poke@ ST,AD,A%(I,3)+A%(I,4)*$10:AD-AD+1:next
 1190 poke@ ST,AD,A%(0,0),A%(0,2)+A%(0,3)*$80,A%(0,4),A%(0,5) an
d $FF,A%(0,6)
 1200 AD-AD+5:next
 1210 '
 1220 '   音色データ
 1230 '
 1240 ' MAIN  MELODY
 1250 data  61, 15,  2,  0,  0,  0,  0,  0,  0,  0
 1260 data  31,  3,  2,  0,  3, 21,  1,  1,  2,  0
```

```
1270 data  31,  0,  1,  7,  3,  4,  1,  2, -3,  0
1280 data  31,  0,  1,  7,  3,  7,  1,  4,  0,  0
1290 data  31,  0,  1,  7,  2,  4,  2,  2,  1,  0
1300 '
1310 ' SUB CHORD
1320 data  61, 15,  2,  0,  0,  0,  0,  0,  0,  0
1330 data  31,  3,  2,  3,  2, 20,  2,  4, -3,  0
1340 data  31,  4,  3,  7,  2,  5,  1,  4,  1,  0
1350 data  31,  4,  3,  7,  2,  5,  1,  4,  0,  0
1360 data  31,  4,  3,  7,  2,  5,  1,  8,  0,  0
1370 '
1380 ' BASS GUITER
1390 data  27, 15,  0,  0, 93,  1,  1,  0,  0,  0
1400 data  31, 15, 14,  9, 12, 45,  2,  0,  0,  2
1410 data  31, 11, 12,  5,  5, 40,  1,  0,  3,  0
1420 data  31,  8,  7,  5,  7, 30,  1,  0,  0,  0
1430 data  31,  6,  5,  7,  3,  0,  0,  1, -2,  1
1440 '
1450 ' SNERE DRUM
1460 data  60, 15,  0,  0,  0,  0,  0,  0,  0,  0
1470 data  30, 18, 11,  6,  0,  7,  0, 15,  3,  0
1480 data  31, 18, 15,  8,  2,  7,  0,  0, -2,  0
1490 data  28, 15, 15,  5,  1,  7,  1,  0,  3,  0
1500 data  28, 15, 15,  5,  1,  0,  1,  0, -3,  0
1510 '
1520 ' TOM-TOM
1530 data  33, 15,  0,  0,  0,  0,  0,  0,  0,  0
1540 data  29, 22,  0, 10,  5,  7,  0,  3,  0,  0
1550 data  31, 20,  0, 10, 10, 15,  0,  3, -2,  0
1560 data  31, 22,  0, 10, 10, 51,  0,  2, -3,  0
1570 data  29, 12,  0,  6, 10,  0,  2,  1,  0,  0
1580 '
1590 ' BASS DRUM
1600 data  60, 15,  0,  0,  0,  0,  0,  0,  0,  0
1610 data  31,  0,  0,  0,  0,  2,  0, 15,  3,  0
1620 data  31, 23, 15, 15,  9,  0,  0,  0, -2,  0
1630 data  31, 17,  5,  6,  5, 17,  0,  0,  3,  0
1640 data  31, 17,  5,  8,  5,  0,  0,  1, -3,  0
1650 '
1660 ' P.S.G.  [Release off]
1670 data  61, 15,  0,  0,  0,  0,  0,  0,  0,  0
1680 data  31,  0,  0,  0,  0, 23,  0,  4,  0,  0
1690 data  31,  0,  0,  0,  0, 15,  0,  2,  0,  0
1700 data  31,  0,  0,  0,  0, 15,  0,  2,  0,  0
1710 data  31,  0,  0,  0,  0, 15,  0,  2,  0,  0
1720 '
1730 '   MML設定・MMLデータ
1740 '
1750 A0$="t156q8l8@v111o4"
1760 B0$="t156q8l8@v123@2o3"
1770 C0$="t156q8l8@v125o3"
1780 D0$="t156q7l8v13o3"
1790 E0$="t156q7l8v13o3"
1800 F0$="t156q8ol116y7,28y6,0sm800"
1810 A1$="@1o3c2.cc&c1c2&cq4cq8rc&c1"
1820 B1$="o4"+string$(16,"<c>c")
1830 C1$="@5"+string$(16,"cr")
1840 D1$="v13o3g2&g&gga1&aa-2&a-q4a-q7rg1&g"
1850 E1$="v13o3e2&e&eef1&ff2&fq4fq7re1&e"
1860 F1$="sm800"+string$(64,"c")
1870 C2$="@5"+string$(12,"cr")+"@5cr@3c4@5cc16c16@3c>@4@V127g16
e16<@V125"
1880 A3$=">c2.cc&c1c2&cq4cq8rc&c1"
1890 C3$=string$(8,"@5cr@3c4")
1900 D3$=">g2&g&gga1&aa-2&a-q4a-q7rg1&g"
1910 E3$=">e2&e&eef1&ff2&fq4fq7re1&e"
1920 A4$="c2.cc&c1dddrrdrc rr@0o5"
1930 B4$=string$(8,"<c>c")+"cccrrcrcrr"
1940 CX$=string$(4,"@5cr@3c4")+"@3ccc>@4g16e16c<@3crc"
1950 C4$=CX$+">@4@V127l16rrrrg32g32ggge32e32eeed32d32ddd<l8@V12
5"
1960 D4$="g2&g&gga1&aa-a-a-rra-rg rrq7v13"
1970 E4$="e2&e&eef1&ffffrrfre rrq7v13"
1980 F4$=string$(32,"c")+"crcrcrrrrcrrrcr rrrrsm800"
1990 A5$="f4.ee2&e2<g4a4>f4.ee2&e2.rr"
2000 B5$="o4"+string$(8,"<c>c")+string$(8,"<b->b-")
2010 C5$=string$(8,"@5cr@3c4")
2020 D5$="o4c&c&c<b>rrd&d&dcrr<b>crrc&c&c<b->rrd&d&dcrr<b->crr"
2030 E5$="o3"+string$(2,"g&g&ggrrg&g&ggrrggrr")
2040 F5$="sm800"+string$(64,"c")
2050 A6$="f4.ee2&e2d4e4f1a-4.g4.f4"
2060 B6$=string$(8,"<a>a")+string$(8,"<f>f")
2070 D6$="c&c&c<b>rrd&d&dcrr<b&b>c&cc2&cded2&dc&cd&d"
2080 E6$="a&a&aarra&a&aarra&aa&af2&fa&aa-2&a-f&ff&f"
2090 C6$=string$(7,"@5cr@3c4")+"@5cr@3c16c16c16c16"
2100 A7$="e1&e4<e4g4>d4d-2.<b-4a2&{agf}2"
2110 B7$=string$(8,"<e>e")+string$(8,"<a>a")
2120 D7$="e1&e2&e&ed&dd-2.&d-&d-<b-&b-&b-&b-b-&b-g&g>"
2130 E7$="g1&g2.&g&ge2.&e&ee&e&e&ee&ee&e"
2140 A8$="e4fd&d2&d4.def4ga2.a-4>f2d2"
2150 B8$=string$(8,"<d>d")+string$(8,"<f>f")
2160 D8$="d2.&d&dc&c&c&cd&d&d&dc2.&c&cd&d&d&df&f&f&f"
2170 E8$="f2.&f&ff&f&f&ff&f&f&ff2.&f&fa-&a-&a-&a-a-&a-&a-&a-"
2180 A9$="e1&e4<e4g4>d4d-1&d-ed-<b-agfe"
2190 A10$="d2.e4{fefgfg}1a2.g>g&g2{ab-b>cd-de-e}2"
2200 B10$=string$(8,"<d>d")+"<f>f<f>f<f>f<f>f<g>g<a->a-<a>a<b>b
"
2210 C10$=string$(7,"@5cr@3c4")+">@4@V127{ggeeddcc}2<@V125"
2220 D10$="d2&d&dd&dc&c&c&cd&d&d&dc2&c&ccg1&g"
2230 E10$="f2&f&ff&ff&f&f&ff&f&f&ff2&f&ffg1&g"
2240 F10$=string$(56,"c")
2250 A11$="d2.e4{fefgfg}1a2.g>g&g1"
2260 B11$=string$(8,"<d>d")+"<f>f<f>f<f>f<f>f<g>g<g>g<g>g<g>g"
2270 A12$="@1o3cc<b>cr2rrr<ab>c4."
2280 B12$=string$(8,"<a>a")
2290 C12$="@5cr16c16cc@3c4rr@5cr16c16ccrc16c16@3c4"
2300 D1X$="v15av14av15av14av15gv14gv15av14ar2"
2310 D12$="q8l16o3"+D1X$+"r4rrv15ev14ev15ev14ev15ev14ev13e4"
2320 E12$="q8l16o3v15ev14ev15ev14ev15dv15dv15ev14er2r1"
2330 F12$="sm800"+string$(8,"crcc")
2340 B13$=string$(8,"<f>f")
2350 D13$=D1X$+"r4rrv15fv14fv15fv14fv15fv14fv13f4"
2360 E13$="v15fv14fv15fv14fv15ev15ev15fv14fr2r1"
2370 A14$="ddcdr2rrr<b>cd4."
2380 B14$=string$(8,"<g>g")
2390 D14$="v15bv14bv15bv14bv15av14av15bv14br2.rrv15gv14gv15gv14
gv15gv14gv13g4"
2400 E14$="v15gv14gv15gv14gv15fv14fv15gv14gr2r1"
2410 A15$="cc<b>cr2r1"
2420 B15$=B12$
2430 D15$=D1X$
2440 E15$=E12$
2450 C15$="@5cr16c16cc@3c4rr@5c16c16c16c16@3c@5c16c16cc@3c16@5c
16c"
2460 F15$=string$(4,"crcc")
2470 A16$="@6o5a1&acdegedc"
2480 D16$=">v15ev14ev15ev14ev15dv14dv15ev14ev7errrr4<r1"
2490 E16$=">v15cv14cv15cv14c<v15bv14b>v15cv14cv7crrrr4<r1"
2500 A17$="r1rg<a>ce<a>cd"
2510 D17$=D16$
2520 E17$=E16$
2530 A18$="r1@1o3rrc4<b4>c4"
2540 D18$=">v15dv14dv15dv14dv15cv14cv15dv14dv7drrrr4"
2550 D18$=D18$+"<r4"+string$(3,"v15gv14gv13g8")
2560 E18$="v15bv14bv15bv14bv15av14av15bv14bv7brrrr4"
2570 E18$=E18$+"r4"+string$(3,"v15dv14dv13d8")
2580 A19$="cc<b>cr2r@v109ff4gg4.@v111@0o5"
2590 B19$=string$(4,"<a>a")+"<rffrgg4.>"
2600 C19$="@5cr16c16cc@3c4rrr@3cc4cc>@4@V127{gged}4<@V125"
2610 D19$=D1X$+"rrv15av14av15av14av9arv15bv14bv15bv14bv13b4q7l8
v13"
2620 E19$="v15ev14ev15ev14ev15dv14dv15ev14er2q7l8v13"
2630 F19$=string$(4,"crcc")+"y6,5m2500c8y6,0m1000crcrrrcrcrr4sm
800"
2640 C20$=CX$
2650 '
2660 '   LFO設定・MML演奏
2670 '
2680 tone lfo 4,1,1,1,-2:LOOP=0
2690 play A0$,B0$,C0$,D0$,E0$,F0$
2700 play A1$,B1$,C1$,D1$,E1$,F1$
2710 play A1$,B1$,C2$,D1$,E1$,F1$
2720 play A3$,B1$,C3$,D3$,E3$,F1$
2730 if LOOP=1 then 3050
2740 play A4$,B4$,C4$,D4$,E4$,F4$
2750 for I=0 to 1
2760 play A5$,B5$,C5$,D5$,E5$,F5$
2770 play A6$,B6$,C5$,D6$,E6$,F5$
2780 play A5$,B5$,C5$,D5$,E5$,F5$
2790 play A6$,B6$,C6$,D6$,E6$,F5$
2800 play A7$,B7$,C5$,D7$,E7$,F5$
2810 play A8$,B8$,C5$,D8$,E8$,F5$
2820 play A9$,B7$,C5$,D7$,E7$,F5$
2830 if I=1 or LOOP=1 then 2860
2840 play A10$,B10$,C10$,D10$,E10$,F10$
2850 next
2860 play A11$,B11$,C10$,D10$,E10$,F10$
2870 if LOOP=1 then 2700
2880 for I=0 to 1
2890 play A12$,B12$,C12$,D12$,E12$,F12$
2900 play A12$,B13$,C12$,D13$,E13$,F12$
2910 play A14$,B14$,C12$,D14$,E14$,F12$
2920 if I=1 then 2950
2930 play A15$,B15$,C12$,D15$,E15$,F12$
2940 next
2950 play A15$,B15$,C15$,D15$,E15$,F15$
2960 for I=0 to 1
2970 play A16$,B12$,C12$,D16$,E16$,F12$
2980 play A17$,B13$,C12$,D17$,E17$,F12$
2990 play A18$,B14$,C12$,D18$,E18$,F12$
3000 if I=1 then 3030
3010 play A15$,B15$,C12$,D15$,E15$,F12$
3020 next
3030 play A19$,B19$,C19$,D19$,E19$,F19$
3040 LOOP=1:goto 2750
3050 play A4$,B4$,C20$,D4$,E4$,F4$
3060 play wait:play init
3070 clear max:end
3080 '
```

日本音楽著作権協会(出)許諾番号　第8872218-801

リスト2 ANGEL[A]

```
1000 '+--------------------------------------+
1010 '|       「ANGEL」    By KYOSUKE HIMURO    |
1020 '+--------------------------------------+
1030 '
1040 '■■ TONE DATA ■■
1050 DEVICE STR$(PEEK(&H74BE)AND15)+":"
1060 IF MEM$(&HAA09,2)=" ウ" THEN VR=400
1070 MEM$(&HB190+VR,36)=HEXCHR$("F9 00 33 20 44 40 16 11 27 00 5
F 5F 5F 56 00 00 00 00 04 04 04 04 05 05 05 05 00 00 00 00 00 00
 80 00 00 00 ")'  1:Guitar
1080 MEM$(&HB1B4+VR,36)=HEXCHR$("D1 00 21 00 00 00 11 1F 16 0F 1
F 1F 1F 5F 00 00 00 03 04 00 00 04 03 03 03 24 00 00 00 00 00 00
```

▶あのウワサのドラスピを買ったけど，やってみると凄いとは思わなかった。なぜだろう？音もよかったし，グラフィックもまあまあよかった。ゲームバランスもよくとれてたと思うのに……，なぜだろう。誰か面白いゲームを教えてくれー。　　釈永　聡（17）富山県

```
 80 00 00 00 ")'  2:Guitar
1090 MEM$(&HB1D8+VR,36)=HEXCHR$("E4 00 31 31 70 71 07 0E 16 00 1
4 14 1F 13 08 0A 08 02 05 0A 05 01 56 A3 56 25 00 00 00 00 00 00
 80 00 00 00 ")'  3:Vocal
1100 MEM$(&HB244+VR,36)=HEXCHR$("E9 00 01 01 01 01 1B 16 15 00 1
4 14 14 54 08 04 08 0A 02 00 00 06 44 44 64 65 00 00 00 00 F4 00
 80 00 00 00 ")'  6:Guitar
1110 MEM$(&HB2F8+VR,36)=HEXCHR$("FA 50 21 31 31 71 16 25 2A 00 D
E DF 5E 4D 01 01 01 02 00 00 00 00 11 15 15 06 00 80 80 80 0C CD
 8A 00 02 00 ")' 11:Keyboard
1120 MEM$(&HB364+VR,36)=HEXCHR$("E0 00 46 45 40 41 1D 2B 19 00 D
F DF 9A 9F 07 06 09 08 07 06 06 04 25 15 15 35 80 00 00 00 00 C8
 80 00 02 00 ")' 14:E Bass
1130 MEM$(&HB6C4+VR,36)=HEXCHR$("FC 00 0C 00 00 00 00 00 02 00 1
F 1C 1E 1F 00 90 13 90 C0 00 0A 00 02 F8 F8 E7 00 00 00 00 F0 C8
 80 00 02 00 ")' 38:Snare
1140 MEM$(&HB70C+VR,36)=HEXCHR$("F8 00 01 0E 00 50 00 00 07 00 1
E 1E 19 1D 1A 1C 10 07 40 C0 40 00 FD FE F8 F8 00 00 00 00 D0 C8
 80 00 00 80 ")' 40:B.Drum
1150 '
1160 RUN "ANGEL[B].mml"
```

リスト3 ANGEL[B].mml

```
1000 ' +----------------------------------------+
1010 ' |          『 ANGEL 』     By  KYOSUKE HIMURO      |
1020 ' +----------------------------------------+
1030 MAXFILES0:DEFINT A-Z:CLS0 : R=0 : FLG=1
1040 TEMPO0 : PLAY"T160" : GOTO 1280
1050 '■■ MML PLAY ■■
1060 LABEL"!":"Drums sub"
1070 'D$="":E$="":F$="":G$=""'A$="":B$="":C$="":G$=""
1080 PLAY    A$;:PLAY":"+B$;:PLAY":"+C$;:PLAY":"+D$;
1090 PLAY":"+E$;:PLAY":"+F$;:PLAY":"+G$;:PLAY":"+H$;
1100 IF FLG=1 THEN PLAY":"+I$;:PLAY":"+I$ ELSE PLAY ""
1110 RETURN
1120 '■■ Drums sub ■■
1130 LABEL "Drums sub"
1140 A=INSTR(A$,"I40") : IF A=0 THEN 1170            ' Bass Drum
1150 A$=LEFT$(A$,A-1)+"i40o1@v127"+MID$(A$,A+3,LEN(A$)-A)
1160 GOTO1140
1170 A=INSTR(A$,"I38") : IF A=0 THEN RETURN          ' Snare Drum
1180 A$=LEFT$(A$,A-1)+"i38o3@V123"+MID$(A$,A+3,LEN(A$)-A)
1190 GOTO 1170
1200 '
1210 ' +-------------------------------------------+
1220 ' | 01:Guitar A     02:Guitar B     03:Vocal     |
1230 ' | 06:Guitar B-2   11:Keyboard     14:Bass      |
1240 ' | 38:Snare Drum   40:Bass Drum                 |
1250 ' +-------------------------------------------+
1260 '
1270 '■■ sub DATA ■■
1280 A3$="I40CI38CI40CI38C"
1290 A1$=STRING$(2,A3$)
1300 A4$="I40CI38CI40C8C8I38C"
1310 A2$=A3$+A4$
1320 I1$="Y6,3C4Y6,0"+STRING$(14,"C")
1330 I2$=STRING$(16,"C")
1340 S$="S1,3,0,10 H1 =0"
1350 '■■ Intro ■■       SPECIAL THANKS TO BALLADE SPORTS CR-X
1360 A$="I40@V120O1 L8  Q7 P3 "+S$
1370 B$="I02@V117O5 L8  Q6 P3 "+S$
1380 C$="I02@V117O5 L8  Q6 P3 "+S$
1390 D$="I01@V113O4 L8  Q6 P3 "+S$
1400 E$="I01@V113O4 L8  Q6 P3 "+S$
1410 F$="I01@V113O2 L8  Q6 P3 "+S$
1420 G$="I14@V117O3 L8  Q6 P3 "+S$
1430 H$="I03@V121O5 L8  Q7 P3 "+S$
1440 I$="Y7,7^1S4,1,0,10 L8 Q5 =2 "
1450 "!"
1460 '■■ A ■■
1470 A$="R1 I40CI38CCI40CI38CCI40CI38C"
1480 B$="P1D+EED+EED+E E4R2."
1490 C$="A+BBA+BBA+B B4"
1500 D$="P2"+B$
1510 E$=C$
1520 F$="R1 I40CI38CCI40CI38CCI40CI38C"
1530 G$=""
1540 H$=""
1550 I$="R1 RY6,0V10CCRCCRC"
1560 "!"
1570 '■■ B ■■
1580 A$="L4I40CI38CR8I40C8I38C8I40C8 CI38CI40C8C8I38C"
1590 B$="P3
1600 C$="
1610 D$="E4E4<E>E<EE >E4E4<E>E<AA"
1620 E$="P3O5B4B4RBRR B4B4RBRR"
1630 F$="I01O5P2"+D$:D$="O4P1"+D$
1640 H$="R4E4E4.R REEEE2"
1650 I$="C4"+STRING$(14,"C")
1660 "!"
1670 D$=">E4E4<A>E4<B >F+4F+4F+F+F+R
1680 E$="A4A4RA4R D+4D+4D+D+D+R
1690 F$=D$
1700 H$="R4C+4D+4E4 G+4.F+F+2"
1710 I$=I2$
1720 "!"
1730 D$=STRING$(2,"L8A&AAARBQ0BL64_1A+_A_G+_G_F+_F_EQ6_D+~8")
1740 E$=STRING$(2,"L8D+&EEEREQ0EL64_1D+_D_C+_C<_B_A+_AQ6_G+>~8")
1750 F$=D$
1760 H$="R4EEE4.R REEEE2"
1770 "!"
1780 A$="I40CI38CR8I40C8I38C8I40C8 CI38CR8C8I40C"
1790 D$="O4L8B&>CC+C+RC+4C+ D+4.D+D+4L64Q0D+8_1D_C+_C<_B_A+_G+_F
+Q6_F~8"
1800 E$="O5L8D+&EEERE4E F+4.F+F+4L64Q0F+8_1G_E_D+_D_C+_C<_BQ6_A~
8"
1810 F$="O5L8G+&AAARA4A B4.BB4L64Q0B8_1A+_G+_F+_F_E_D+_DQ6_C+~8"
1820 H$="R4C+4D+4E4 G+4.F+F+2"
1830 I$=STRING$(15,"C")+"4"
1840 "!"
1850 A$="I38G8E8C8I40CR4. I38@V126C8C8C4"
1860 B$="O4@V111L8C+<G+BA+&A+2 BBL64Q0B8B>C+DD+Q6E
1870 D$=LEFT$(B$,21)+" BBBL64A+AG+GF+FED"
1880 E$="R1 O4F+F+F+L64ED+DC+C<BA+A"
1890 F$="R1 O2I38@V126C8C8C4"
1900 G$="R4.O2F+8&F+2 B8B8B4"
1910 H$="R1 R2"
1920 I$="CR4Y6,3C4.Y6,0"
1930 "!"
1940 '■■ C ■■
1950 A$=A1$
1960 B$="O4L8I06Q4@V109"+STRING$(16,"E")
1970 C$="O4L8I06Q4@V109"+STRING$(16,"B")
1980 D$="O3P3K6L8@V115EE>ER<E>E4<E >G+<EE>G+<EE>G+<E"
1990 E$="O3P3K3L8@V115EE>BR<E>B4<E >B <EE>B <EE>B <E"
2000 F$="P3"
2010 G$="O3L8EEEEEEEE D+&EEEEED+E"
2020 H$="R4O4B4B4G+4 E16&E.RF+F+G+AF+&"
2030 I$=I1$
2040 "!"
2050 A$=A2$
2060 B$=RIGHT$(B$,14)+"I02Q0E8L64_1D+_D_C+_C<_B_A+_AQ4_G+>~8"
2070 C$=RIGHT$(C$,14)+"I02Q0B8L64_1A+_A_G+_G_F+_F_EQ4_D+~8"
2080 D$="R4O4ERRER<E >BRRBR44"
2090 E$="R4O4BRRBR<E >ERRER4E4"
2100 G$=RIGHT$(G$,12)+" D+EE<B>EEE<B"
2110 H$="F+4REEEEE D+EF+ER2"
2120 I$=I2$
2130 "!"
2140 A$=A1$
2150 B$="I06L8"+STRING$(16,"C+")
2160 C$="I06L8"+STRING$(16,"G+")
2170 D$="R4BRRBR4 BRRBR4BR"
2180 E$="R4>ERRER4 D+RRD+R4D+R"
2190 G$="O3C&"+STRING$(15,"C+")
2200 H$="R4BB4BA4 G+4RB>C+C+4<B"
2210 "!"
2220 A$=A2$
2230 B$=RIGHT$(B$,28)+"I02Q0C+8L64_1C<_B_A+_A_G+_G_F+Q4_F>~8"
2240 C$=RIGHT$(C$,28)+"I02Q0G+8L64_1G_F+_F_E_D+_D_C+Q4_C~8"
2250 D$="R4G+RRG+R4 G+RRF+&G+B4."
2260 E$="R4C+RRC+R4 C+RRF+&G+B4."
2270 G$=RIGHT$(G$,16)+"C&C+C+<G+>C+C+C+<G+"
2280 H$="G+4R4.EE4 F+EF+E4.R4"
2290 "!"
2300 A$=A1$
2310 B$="O3I06L8AAAAAA16A16AA AAAAAA>I02AB"
2320 C$="O4I06L8EEEEEE16E16EE EEEEEE>I02EF+<"
2330 D$="O3AA>ARRAR4 >ERR<BR4BR"
2340 E$="O3AA>>C+RRC+R4 ARRER4ER"
2350 G$=STRING$(15,"A")+"B"
2360 H$="R4B4EEEE BB4E4.R4"
2370 I$=I1$
2380 "!"
2390 A$=A2$
2400 B$="I06"+STRING$(14,"C")+">I02Q0C8L64<_1B_A+_A_G+_G_F+_FQ4_
E~8"
2410 C$="I06"+STRING$(14,"G")+">I02Q0G8L64_1F+_F_E_D+_D_C+_C<Q4_
B>~8"
2420 D$="R4GRRGR4 R4.GR4G4"
2430 E$="R4CRRCR4 R4.CR4C4"
2440 G$="O3"+STRING$(15,"C")
2450 H$="R4G4G4A4 B4>C4C4C4"
2460 I$=I2$
2470 "!"
2480 B$="O4I06L8GGGGG>D<GG GGGGGGGG"
2490 C$="I06L8DDDDDGDD DDDDDDDD"
2500 D$="
2510 E$="
2520 G$="O2"+STRING$(16,"G")
2530 H$="<B4.G4D4. R2RDDB"
2540 I$=I1$
2550 "!"
2560 A$=A3$+" I40CI38CC8I40C8C8I38C8"
2570 B$="A16A16AAAAAI02Q0A8L64_1G+_G_F+_F_E_D+_DQ4_C+~8L8 <I06BB
BBB>BI02Q0B8L64_1A+_A_G+_G_F+_F_EQ4_D+~8"
2580 C$="D16D16DDDDDI02Q0D8L64_1C+_C<_B_A+_A_G+_GQ4_F+~8L8 I06F+
F+F+F+F+>F+I02Q0F+8L64_1F_E_D+_D_C+_C<_BQ4_A+~8"
2590 D$="
2600 E$="
2610 G$="O3DDDDDDDC <BBBBBBBB"
2620 H$="A4.>D4.<A4 R1"
2630 I$="CCCCCCCC C4C4C4RY6,3C&"
2640 "!"
2650 '■■ C'■■
2660 A$=A1$
2670 B$="L8I06P3K5O4"+STRING$(16,"E")
2680 C$="L8I06P3K5O4"+STRING$(16,"B")
2690 D$="O4Q6@V115EEBREB4E BEEBEEBR"
2700 E$="O4Q6@V115EE>ER<E>E4<E >D+<EE>D+<EE>D+R"
2710 F$=""
2720 G$="O3@V117EEEEEEEE D+&EEEEEEE"
2730 H$="R4I03P3K5@V121Q7O4B4B&AAG+& G+4RF+F+G+AG+"
```

▶2月号168ページの川村さん，「ライブマン」も凄いかもしれないけれど「サイバーコップ」も凄いですよ。これを見ると，日本の近未来のコンピュータ市場は「TRON」ではなく，ラップトップから都庁の大型コンピュータまで「富士通」に制圧されていることがよーくわかります。
畦地　真一郎 (18) 北海道

```
2740 I$="Y6,0"+I2$
2750 "!"
2760 A$=A2$
2770 B$=RIGHT$(B$,14)+"I02E4"
2780 C$=RIGHT$(C$,14)+"I02B4"
2790 D$="R4O5BRR<BR4 >D+ED+EF+ED+E"
2800 E$=LEFT$(D$,11)
2810 F$="R1 L8RI11P1K5Q6@V124O5ED+EF+E<B>E"
2820 G$=RIGHT$(G$,10)+" D+&EEEEEE<B"
2830 H$="REEEEEEE F+ED+E4.R4"
2840 I$=I2$
2850 "!"
2860 A$=A1$
2870 B$="I06C+C+C+C+C+G+4"+STRING$(9,"C+")
2880 C$="I06G+G+G+G+G+>C+4<"+STRING$(9,"G+")
2890 D$="O4R4BRRBR4 BRRBR4G+4"
2900 E$="O5R4ERRER4 D+RRD+R4<B4"
2910 F$=""
2920 G$="O3"+STRING$(16,"C+")
2930 H$="R4B4B4A4 G+4.G+>C+.C+16<B4"
2940 "!"
2950 B$=STRING$(14,"C+")+"I02C+4"
2960 C$=STRING$(14,"G+")+"I02G+4"
2970 D$="R4G+RRB4."+STRING$(8,"G+")
2980 E$="R4>C+<RRB4."
2990 F$="R1 P2RED+EF+E<B>E"
3000 G$=RIGHT$(G$,22)+"<G+>C+C+C+<G+"
3010 H$=">C+16&<G+.REEEEE F+ED+E&<G+4R4"
3020 "!"
3030 B$="I06O3"+STRING$(14,"A")+"I02A16&A+16&B"
3040 C$="I06O4"+STRING$(14,"E")+"I02D+16&E16&F"
3050 D$="O3AA>ER<A>E4<A >ERRER4EE"
3060 E$="O3AA>AR<A>A4<A >BRRBR4BB"
3070 F$="R1 R2RO4P3B>EG"
3080 G$=STRING$(15,"A")+"B"
3090 H$="O4RBB4E4EE BB4E4.R4"
3100 I$=I1$
3110 "!"
3120 A$=A3$+" I40C8C8I38CI40CI38C"
3130 B$="I06O4"+I2$
3140 C$="I06"+STRING$(16,"G")
3150 D$="R4GRRG4R GR4.R2"
3160 E$="R4>CRRC4R CR4.R2"  ,
3170 F$=""
3180 G$="O3"+I2$
3190 H$="R4G4G4A4 B4>C4C4C4"
3200 I$=I2$
3210 "!"
3220 A$=A2$
3230 B$="O4"+STRING$(14,"G")+"I02G4"
3240 C$="O5"+STRING$(14,"D")+"I02D4"
3250 D$="R4O5GRR2 RG4R4.G4"
3260 E$="R4O5DRR2 RD4R4.D4"
3270 F$="R4@V112I01O4BRR2 RB4R4.B4"
3280 G$=STRING$(2,"O2GG>D<GGG>A<G")
3290 H$="<B4.G4D4. R2DD4A"
3300 "!"
3310 A$=A3$+" I40CI38@V125C8.C16C8C8C8C8"
3320 B$="I06DDDDDDI02D4 I06F+F+F+F+F+F+F+F+"
3330 C$="I06O4AAAAAAI02A4 I06BBBBBBBB"
3340 D$="R4ARRA4R D+D+D+D+G+2"
3350 E$="R4F+RRF+4R <BBBBB2"
3360 F$="R4DRRD4R RO4I11@V124AA+BR>B4."
3370 G$=">DD<A>DDD<A>D <BBBF+BBF+B"
3380 H$="A4.>D4&<AA4 RF+4F+R2"
3390 "!"
3400 '■ D ■
3410 A$=STRING$(2,A4$)
3420 IF R=1 OR R=4 OR R=7 THEN A$=A4$+"8I40C8 CI38CI40C8C8I38C"
3430 B$="I02P1O5"+STRING$(2,"D+&EEREEEE")
3440 C$="I02P2O5"+STRING$(2,"F+16&G16&G+G+RG+G+G+G+")
3450 D$=STRING$(2,"Q3O4A16&A+16&BBRBBBB")
3460 E$=STRING$(2,"Q3O5D+EERED+ED+")
3470 F$="I02K0@V109O4"+STRING$(2,"B4G+RBBG+G+")
3480 G$="@V121L8"+STRING$(2,"O2E4E4EB>C+<B")
3490 H$="R4O5E4E4.R REEEE2"
3500 IF R=1 OR R=4 OR R=7 THEN H$="R4O5EEE2 REEEE2"
3510 I$=I1$
3520 IF R=1 OR R=4 OR R=7 THEN I$=I2$
3530 "!"
3540 IF R=1 OR R=4 OR R=7 THEN A$=A4$+" I40CI38@V126C8.C16C8C8C8
C8"
3550 B$=RIGHT$(B$,10)+"D+4D+4D+D+G+F+"
3560 C$=RIGHT$(C$,22)+"F+4F+4F+F+ED+"
3570 D$=RIGHT$(D$,18)+"B4BBBBBB"
3580 E$=RIGHT$(E$,13)+"D+4D+D+ED+ED+"
3590 F$=RIGHT$(F$,11)+"B4B4BBBB"
3600 G$="A4A4AEF+B BBBBBF+Q0G+8L64GF+FED+DC+Q6C"
3610 IF R=1 OR R=4 OR R=7 THEN G$="A4AAAEF+E F+BF+BG+"+RIGHT$(G$
,23)
3620 H$="R4C+4D+4E4 G+4.F+F+2"
3630 IF R=1 OR R=4 OR R=7 THEN H$=LEFT$(H$,19)+"4.<G+8"
3640 I$=STRING$(14,"C")+"Y6,3C4Y6,0"
3650 IF R=1 OR R=4 OR R=7 THEN I$=STRING$(12,"C")
3660 "!"
3670 R=R+1:IF R=1 OR R=4 OR R=7 THEN 3410
3680 '■ E ■
3690 A$=STRING$(2,A4$)
3700 B1$="C+8&L64_1C_<B_A+_G+_F+_F_E_D+¯8L8"
3710 B$="¯5O4C+2C+C+"+B1$+" F+2>F+BC+C+"
3720 C1$="G+8&L64_1G_F+_F_E_D+_D_C+_C¯8L8"
3730 C$="¯5O4G+2G+G+"+C1$+" C+2RRG+G+"
3740 D$="@V113O4C+C+C+4C+C+C+4 <F+F+F+4F+F+F+4>":DD$=D$
3750 E$="@V113O4G+G+G+4G+G+G+4 C+C+C+4C+C+C+4":EE$=E$
3760 F$="O5C+2C+C+C+4 R1"
3770 G$="L8O3C+4C+4C+C+C+C+"+STRING$(8,"F+")
3780 H$="O5FFFFFFD+F F+F+F+FD+C+C+4"
3790 I$=I1$
3800 "!"
3810 B$="C+4C+4C+C+"+B1$+" F+F+>A+4<G+G+>C4"
3820 C$="G+4G+4G+G+"+C1$+" C+C+>C+4<G+G+>D+4"
3830 D$=LEFT$(D$,29)+"G+G+G+4"
3840 E$=LEFT$(E$,28)+"D+D+D+4"
3850 F$="C+4C+4C+C+C+4 F+F+F+4<D+D+>G+4"
3860 G$=STRING$(8,"C+")+"F+F+F+F+G+G+G+G+"
3870 H$="RFFFFFFF C+C+C+D+4.R4"
3880 I$=I2$
3890 "!"
3900 A$=A4$+" I40CI38CI40CI38@V126C8C16C16"
3910 B$="O4C+4C+2"+B1$+" F+1P3_5"
3920 C$="O4G+4G+2"+C1$+" C+1P3_5"
3930 D$=DD$
3940 E$=EE$
3950 F$="O5C+4C+2C+4 <F+1"
3960 G$=STRING$(8,"C+")+"F+F+F+F+F+F+F+C+"
3970 H$="FFFFFFD+F F+F+F+FD+C+C+4"
3980 I$=STRING$(14,"C")
3990 "!"
4000 IF R=5 THEN 5140
4010 IF R=8 THEN 5360   '■ to Coda ■
4020 '■ E1 ■
4030 A$=A1$
4040 B$="I06"+STRING$(14,"B")+"I02BB"
4050 C$="I06"+STRING$(14,"F+")+"I02F+F+"
4060 D$=STRING$(14,"F+")+"4F+"
4070 E$=STRING$(14,"B")+"4B"
4080 F$=B$
4090 G$=STRING$(2,"<BBF+BBBF+B>")
4100 H$="D+2.R4 R4.C+C+D+4C+"
4110 I$=I1$
4120 "!"
4130 A$=A3$+" I40CI38C @V127C8C8C8C8"
4140 B$="I06"+STRING$(10,"G+")+"I02G+8&L64_1G_F+_F_E_D+_D_C+_C¯8
L8I06BBB>I02F+"
4150 C$="I06"+STRING$(10,"D+")+"I02D+8&L64_1D_C+_C<_B_A+_A_G+_G¯
8L8>I06F+F+F+I02>F+"
4160 D$=STRING$(8,"D+")+"D+16D+16D+4D+D+D+D+"
4170 E$=STRING$(8,"G+")+"G+16G+16G+4G+G+G+G+"
4180 F$=B$
4190 G$=STRING$(3,"G+G+>D+<G+")+"BBB<F+"
4200 H$="C4.<G+D+4R4 R1"
4210 I$=STRING$(10,"C")+"Y6,3C4"
4220 "!"
4230 '■ F ■
4240 A$=A2$
4250 B$="O4G+4.&F+F+4.&G+& G+G+A4AB4B"
4260 C$="O5K3"+RIGHT$(B$,26)
4270 D$="O4"+STRING$(16,"E")
4280 E$="O4"+STRING$(16,"B")
4290 F$="K7"+C$:B$="P2K8"+B$
4300 G$="O3"+STRING$(16,"E")
4310 H$="P1K2I02@V109Q4"+RIGHT$(B$,28)
4320 I$="C4Y6,1"+STRING$(14,"C")
4330 "!"
4340 B1$="A4AG+4G+A4 A4.&L64_1G+_G_F+_F_E_D+_D_C+¯8L8R"
4350 B$=B1$+"B4B&"
4360 C$=B1$+"E4E&"
4370 E$=STRING$(16,"A")
4380 F$=C$
4390 G$=E$
4400 H$=B$
4410 I$=I2$
4420 "!"
4430 B$="BA16&B16&A4&A&B16&A16G+&A G+&A&B&AG+16&A16&G+EE&"
4440 C$="E"
4450 D$=STRING$(16,"B")
4460 E$="O5"+STRING$(16,"E")
4470 F$=C$
4480 G$="O3"+STRING$(16,"E")
4490 H$=B$
4500 "!"
4510 A$=A3$+" I40CI38C8C16C16I40C8I38C8C8"
4520 B$="E4>ED<BA16&A+16&A16G16E AGA32&B.&B32B4B>D"
4530 C$=LEFT$(B$,38)+">EG"
4540 D$="EEEEEEEE F+F+F+F+F+F+F+R"
4550 E$="O4AAAAAAAA BBBBBBBR"
4560 F$=C$
4570 G$="AAAAAAAABBBBBBBR"
4580 H$=B$
4590 I$="CCCCCCCC C4R2Y6,3C4"
4600 "!"
4610 '■ G ■
4620 A$=A1$
4630 B$="=1D4=0C16&D16&C<CG4D+16&D16 DRGA&BAG<A"
4640 C$=B$
4650 D$=STRING$(16,"D")
4660 E$=STRING$(16,"G")
4670 F$=C$
4680 G$=STRING$(16,"G")
4690 H$=B$
4700 I$="C4Y6,0"+STRING$(14,"C")
4710 "!"
4720 A$=A2$
4730 B$="O3Q2L16"+STRING$(7,"¯1AAAA")+"A4_7"
4740 C$="O4Q2L16"+STRING$(3,"¯1EEEE")+"¯F+F+F+F+ ¯GGGG¯AAAA¯BBBB
B4_7"
4750 D$="L16E8EEE8"+STRING$(18,"E")+"E8E8E4"
4760 E$="L16A8AAA8"+STRING$(18,"A")+"A8A8A4"
4770 H$=B$
4780 G$=STRING$(16,"A")
4790 F$="R1 R4L16"+STRING$(2,"¯1EEEE")+"E4_2"
4800 I$=STRING$(15,"C")+"Y6,3C"
4810 "!"
4820 A$="I40CI38C8I40CR8I38C I40CI38C8I40CR8I38C"
4830 B$="L8Q4O4G>DC16&D16&C<BG4>D& EEREEER>D+&"
4840 C$="L8Q4O4G>GC16&D16&C<BG4>G& AARAAAR>D+&"
4850 D$="D8DDD8DDD8DDD8DD E8EEE8EEE8EEE8EE"
4860 E$="G8GGG8GGG8GGG8GG A8AAA8AAA8AAA8AA"
```

▶最近，NTTのキャプテンを見かけないという声をよく耳にしますが，レトロの街・函館にはあります。五稜郭の丸井今井デパート7階にあるその端末は，なんと無料でプリントアウトも自由です。ただ，残念なのは端末がランク2(低密度端末)だということです。
神生 総一 (23) 北海道

```
4870 F$=C$
4880 G$="GGGGGGGG AAAAAAAA"
4890 H$=B$
4900 I$="C4Y6,0"+STRING$(14,"C")
4910 "!"
4920 A$="I40CI38C8I40CR8I38C8C16C16 C8F8C8I40C4.R8I38C16C16"
4930 B$="D+8L64DC+C<BA+AG+GF+FED+DC+C<B>>L8B&B4.&>C+& C+1&"
4940 C$=B$
4950 D$="O3"+STRING$(16,"B")+">L8C+<G+BA+&A+2&"
4960 E$="O4"+STRING$(16,"F+")+"L8C+<G+BA+&A+2&"
4970 F$=C$
4980 G$="BBBBBBBB R>G+<BF+&F+2&"
4990 H$=B$
5000 I$="CCCCCCCR R4.Y6,3C4.Y6,0"
5010 "!"
5020 A$="C8C8C8C8C8C16C16C I40C8C8C"
5030 B$="C+8L64C_1<B_A+_A_G+_G_F+_F_E_D+_D_2C+_C_<B_A+_AL8R'R2¯2
0"
5040 C$=B$
5050 D$="A+1 >F+F+F+4"
5060 E$="A+1 BBB4"
5070 F$="R1 O2I38@V126C8C8C4"
5080 G$="F+2.>L64Q0C+C<BA+AG+GF+FED+DC+C<BA+AQ6G+' O2L8BBB4"
5090 H$=B$
5100 I$=""
5110 "!"
5120 R=R+1:GOTO2660
5130 '■■ E2 ■■
5140 A$=A1$
5150 B$="I06O4F+4"+STRING$(12,"F+")+"I02F+4"
5160 C$="I06O4B4"+STRING$(12,"B")+"I02B4"
5170 D$=MID$(B$,4,29)+"F+4"
5180 E$=MID$(C$,4,16)+"B4"
5190 F$="I06O3"+RIGHT$(C$,19)
5200 G$=STRING$(4,"O2BBF+B")
5210 H$="D+1 R4.D+C+D+4C+"
5220 I$=I1$
5230 "!"
5240 A$=A3$+"8C8 C16C16L8CCC @V127CCCCL4"
5250 B$="I06"+STRING$(10,"D+")+"I02D+D+¯5F+F+F+F+_5"
5260 C$="I06"+STRING$(10,"G+")+"I02G+G+¯5BBBB_5"
5270 D$=STRING$(12,"D+")+"¯5F+F+F+F+_5"
5280 E$=STRING$(12,"G+")+"¯5BBBB_5"
5290 F$=C$
5300 G$=STRING$(3,"G+G+>D+<G+")+"¯5G+BBB_5"
5310 H$="C4.<G+D+2 R1"
5320 I$="CCCCCC"
5330 "!"
5340 R=R+1:GOTO3410
5350 '■■ Coda ■■
5360 A$=A4$+" I40CI38C8I40CC8I38C8"
5370 B$="O4G+4G+G+G+G+4G+ C+4C+C+G+G+G+G+"
5380 C$="P1O5C+4C+C+C+C+4C+ <F+4F+F+>C+C+C+C+"
5390 D$="O4R4G+R4G+4G+ RC+4R4G+G+G+"
5400 E$="O5R4C+R4C+4C+ R<F+4>R4C+C+C+"
5410 F$="P2O4"+RIGHT$(C$,32)
5420 G$="O3C+C+C+C+C+C+C+C+ <F+F+F+F+G+>G+D+&C+"
5430 H$="RO5FFFFFFF C+C+C+D+4.RC+"
5440 I$=I2$
5450 "!"
5460 '■■ E'■■
5470 A$=STRING$(2,A4$)
5480 B$="G+4G+4G+G+G+ C+2.Q2C+8C+16C+16Q4"
5490 C$="C+4C+4C+C+C+ <F+2.Q2F+8F+16F+16Q4>"
5500 D$="G+4G+4RG+RG+ C+C+C+4C+C+C+4"
5510 E$="C+4C+4RC+RC+ <F+F+F+4F+F+F+4>"
5520 F$=C$
5530 G$="C+C+C+C+C+C+C+C+ <F+F+F+F+F+F+F+>C+"
5540 H$="FFFFFFD+F F+F+F+FD+&C+C+4"
5550 "!"
5560 A$=STRING$(2,A4$)
5570 B$="G+4G+4G+G+G+4 C+4C+4D+D+G+G+"
5580 C$="C+4C+4C+C+C+4 <F+4F+4G+G+>C+C+"
5590 D$="G+G+G+4G+G+G+4 C+C+C+4D+D+D+D+"
5600 E$="C+C+C+4C+C+C+4 <F+F+F+4G+G+G+G+>"
5610 F$=C$
5620 G$="C+C+C+C+C+C+C+C+ <F+F+F+F+G+G+G+G+"
5630 H$="RFFFFFFF C+C+C+D+4.R4"
5640 "!"
5650 '■■ H ■■
5660 A$=STRING$(2,A4$)
5670 B$="@V115K3Q7O5F&F+G+F&F+>C+<F&F+ >C+<G+>C+C+<F&F+>C+<F&"
5680 C$="@V115K7Q7R32"+MID$(B$,5,42)+"@18&":B$="P1"+B$:C$="P2"+C
$
5690 D$="@V110O4Q3G+G+G+4G+G+G+G+ F+F+F+4F+F+F+F+"
5700 E$="@V110O5Q3C+C+C+4C+C+C+C+ <BBB4BBBB>"
5710 F$="@V115K5Q7O5F&F+G+F&F+>C+<F&F+ >C+<G+G+G+F&F+>C+<F&"
5720 G$="O3C+C+C+C+C+C+C+C+< F+F+F+F+F+F+F+F+>"
5730 H$="@V110I01O4"+RIGHT$(E$,28)
5740 "!"
5750 IF R=9  THEN 6270
5760 A$=A4$+" I40CI38C8I40CC8I38C"
5770 B$="F+>C+<F&F+>C+<G+>C+C+ <F&F+>C+<F&F+G+A+&G+"
5780 C$="F32"+B$+"@18"
5790 D$="D+D+D+4D+D+D+D+ C+C+C+4D+D+D+D+"
5800 E$="<G+G+G+4G+G+G+G+ F+F+F+4G+G+G+G+>"
5810 F$="F+>C+<F&F+>C+<G+G+G+ F&F+G+F&F+G+A+&G+"
5820 G$="C+C+C+C+C+C+C+C+ <F+F+F+F+G+>G+A+&G+"
5830 H$=E$
5840 I$=STRING$(14,"C")+"Y6,3C4Y6,0"
5850 "!"
5860 A$=STRING$(2,A4$)
5870 B$=">D+16=1F2&F8&=0F8.D+16&F16& D+C+<A+G+>C+C+C+C+"
5880 C$=B$
5890 D$="C+C+C+4C+C+C+C+ G+G+G+4G+G+G+G+"
5900 E$="<F+F+F+4F+F+F+F+ >C+C+C+4C+C+C+C+"
5910 F$=LEFT$(B$,37)+"G+G+G+G+"
5920 G$="C+C+C+C+C+C+<A+>C+ <F+F+F+F+F+F+F+F+>"
5930 H$=E$
5940 I$=I2$
5950 "!"
5960 A$=A4$+" I40CI38C8I40CC8I38C8C16C16"
5970 B$="O5K5Q2B16&C+16C+G+G+G+G+G+G+ G+G+G+G+G+G+G+C+"
5980 C$="O5K5Q2F+16&G+16"+STRING$(15,"G+")
5990 D$="G+G+G+4G+G+G+4 C+C+C+4D+D+D+4":DD$=D$
6000 E$="C+C+C+4C+C+C+4 <F+F+F+4G+G+G+4>":EE$=E$
6010 F$="O5Q2F+16&G+16>"+STRING$(15,"C+")
6020 G$="C+C+C+C+C+C+C+C+ <F+F+F+F+G+G+G+G+>"
6030 H$=E$
6040 I$="CCCCCCCR Y6,3C4Y6,0CCCCCC"
6050 "!"
6060 A$="\"+STRING$(2,A4$)
6070 B$="K3Q7O6G+&A+&G+&A+G+&A+G+&A+ G+&A+G+=1C+4.=0C+<E"
6080 C$="K7R32"+RIGHT$(B$,45)+"@18"
6090 D$="G+G+G+4G+G+G+G+ C+C+C+4C+C+C+C+"
6100 E$="C+C+C+4C+C+C+C+ <F+F+F+4F+F+F+F+>"
6110 F$="K5"+RIGHT$(B$,45)
6120 G$="C+C+C+C+C+C+C+C+ <F+F+F+F+F+F+F+F+>"
6130 F$=E$
6140 I$=I2$
6150 "!"
6160 A$=A4$+" I40C8I38C8I40C8C8I38C8I40C8I38C"
6170 B$=">G+&A+G+&A+&G+&A+G+&A+& G+&A+&>C+=1C+4.&=0C+4"
6180 C$=B$
6190 D$=DD$
6200 E$=EE$
6210 F$=B$
6220 G$="C+C+C+C+C+C+C+C+ <F+F+F+F+G+>D+E4"
6230 H$=E$
6240 I$="CY6,1C4.C4C4Y6,3C4Y6,0CCCCCC"
6250 "!"
6260 R=R+1:GOTO 5660
6270 PLAY"R:R:R:R:R:R:R:R"
6280 END
6290 ' +-------------------------------------------+
6300 ' |    1989/01/02 21:21:42  VIP000 CHACHA     |
6310 ' +-------------------------------------------+
```

リスト4 Moonlight Feels Right

日本音楽著作権協会(出)許諾番号　第8872218-801

```
10 /*
20 /*        M O O N L I G H T   F E E L S   R I G H T
30 /*
40 /* Y u k i h i r o   T a k a h a s h i   V e r s i o n
50 /*
60 /*            Music by Bruce Blackman
70 /*
80 /*           Programed by Masahiro Ando
90 /*
100 /*
110 /*    S O U N D   D A T A
120 /*
130 dim char SDRUM(4,10)={
140       60, 15,  0,  0,  0,  0,  0,  0,  0,  3,  0,
150       31,  0,  0,  0,  0,  0,  0, 15,  3,  3,  0,
160       31, 18, 17,  8,  0,  0,  0,  0,  7,  0,  0,
170       31, 24,  1,  7, 15, 10,  0,  0,  0,  0,  0,
180       31, 17, 10,  8, 15,  0,  0,  0,  0,  0,  0}
190 m_vset(71,SDRUM)
200 dim char BDRUM(4,10)={
210       36, 15,  0,  0,  0,  0,  0,  0,  0,  3,  0,
220       31,  1, 27,  9,  0, 20,  0,  0,  7,  0,  0,
230       31,  3, 15,  9,  0,  0,  0,  0,  7,  0,  0,
240       31, 22,  4,  9, 15, 18,  0,  0,  3,  0,  0,
250       31, 16, 14,  9, 15,  0,  0,  0,  3,  0,  0}
260 m_vset(72,BDRUM)
270 dim char HHCL(4,10)={
280       58, 15,  0,  0,  0,  0,  0,  0,  0,  2,  0,
290       31,  0,  0,  0, 12, 18,  0, 15,  3,  1,  0,
300       31,  0,  0,  0, 10, 35,  0, 10,  3,  3,  0,
310       31,  0,  0,  0, 10, 17,  0, 15,  0,  3,  0,
320       31, 16, 25,  7, 14, 12,  0,  1,  6,  0,  0}
330 m_vset(73,HHCL)
340 dim char HHOP(4,10)={
350       58, 15,  0,  0,  0,  0,  0,  0,  0,  2,  0,
360       31,  0,  0,  0, 12, 18,  0, 15,  3,  1,  0,
370       31,  0,  0,  0, 10, 35,  0, 10,  3,  3,  0,
380       31,  8,  0,  0, 10, 17,  0, 15,  0,  3,  0,
390       31, 12, 25,  6, 14, 14,  0,  1,  6,  0,  0}
400 m_vset(74,HHOP)
410 dim char CYMBAL(4,10)={
420       52, 15,  0,  0,  0,  0,  0,  0,  0,  3,  0,
430       31,  0,  0,  0,  0,  7,  0, 15,  6,  0,  0,
440       31, 25, 12,  6,  2,  0,  0, 13,  2,  0,  0,
450       31, 25,  0,  2,  0,  8,  0, 10,  2,  0,  0,
460       31, 25, 13,  6,  4,  0,  0, 15,  6,  0,  0}
470 m_vset(75,CYMBAL)
480 dim char EBASS(4,10)={
490       48, 15,  0,  0,  0,  0,  0,  0,  0,  3,  0,
500       30, 14,  8,  7, 11, 23,  2,  0,  3,  0,  0,
510       24, 10,  8,  7, 11, 63,  3,  0,  3,  0,  0,
520       28,  4,  8,  7, 11, 23,  3,  0,  3,  0,  0,
530       28,  5,  8,  7, 11,  0,  3,  0,  3,  0,  0}
540 m_vset(76,EBASS)
550 dim char EGUITAR(4,10)={
560       57, 15,  0,  0,  0,  0,  0,  0,  0,  3,  0,
```

MOONLIGHT FEELS RIGHT by BRUCE BLACKMAN
©1975 by LOWERY MUSIC CO.,INC./BROTHER BILLS MUSIC
The rights for Japan assigned to FUJIPACIFIC MUSIC INC.

```
 570        31,  7,  4,  7, 15, 41,  3,  4,  7,  0,  0,
 580        31,  4,  4,  9, 15, 35,  3,  6,  0,  0,  0,
 590        31,  4,  4,  9, 15, 43,  3,  6,  3,  0,  0,
 600        31, 10,  3,  9,  0,  0,  2,  2,  0,  0,  0}
 610 m_vset(77,EGUITAR)
 620 dim char ABRASS(4,10)={
 630        60, 15,  0,  0,  0,  0,  0,  0,  0,  3,  0,
 640        10,  7,  0,  7, 11, 31,  6,  4,  3,  0,  0,
 650        18, 15,  0,  7,  0,  0,  0,  4,  7,  0,  0,
 660        10,  6,  0,  7, 10, 26,  0,  4,  7,  0,  0,
 670        18, 15,  0,  7,  0,  0,  0,  4,  3,  0,  0}
 680 m_vset(78,ABRASS)
 690 dim char ASTRINGS(4,10)={
 700         4, 15,  0,  0,  0,  0,  0,  0,  0,  3,  0,
 710        18,  2,  0,  2,  2, 26,  1,  4,  7,  0,  0,
 720        14,  6,  0,  8,  1,  0,  0,  4,  7,  0,  0,
 730        18,  2,  0,  2,  2, 26,  1,  4,  3,  0,  0,
 740        14,  6,  0,  8,  1,  0,  0,  4,  3,  0,  0}
 750 m_vset(79,ASTRINGS)
 760 dim char BACKING1(4,10)={
 770         4, 15,  0,  0,  0,  0,  0,  0,  0,  3,  0,
 780        28, 10,  6,  8, 10,  6,  2,  4,  0,  0,  0,
 790        28, 15,  4,  8,  0,  0,  0,  8,  0,  0,  0,
 800        28, 10,  6,  8, 10,  6,  2,  3,  0,  0,  0,
 810        28, 15,  4,  8,  0,  0,  0,  6,  0,  0,  0}
 820 m_vset(80,BACKING1)
 830 dim char BACKING2(4,10)={
 840        60, 15,  0,  0,  0,  0,  0,  0,  0,  3,  0,
 850        28, 10,  6,  8, 10, 25,  2,  8,  3,  0,  0,
 860        28, 15,  4,  8,  0,  0,  0,  4,  3,  0,  0,
 870        28, 10,  6,  8, 10, 10,  2,  8,  7,  0,  0,
 880        28, 15,  4,  8,  0, 10,  0,  4,  7,  0,  0}
 890 m_vset(81,BACKING2)
 900 dim char EPIANO(4,10)={
 910         4, 15,  0,  0,  0,  0,  0,  0,  0,  3,  0,
 920        28, 10,  2,  4, 10, 26,  2,  3,  3,  0,  0,
 930        28,  6,  1,  6,  9,  0,  2,  3,  7,  0,  0,
 940        28, 10,  2,  4, 10, 26,  2,  4,  3,  0,  0,
 950        28,  6,  1,  6,  9,  0,  2,  4,  7,  0,  0}
 960 m_vset(82,EPIANO)
 970 dim char MARIMBA(4,10)={
 980        60, 15,  0,  0,  0,  0,  0,  0,  0,  3,  0,
 990        31, 23,  9,  3, 14, 25,  1,  6,  7,  0,  0,
1000        31,  8,  9,  4,  5,  0,  2,  1,  7,  0,  0,
1010        31, 12,  5,  2, 15, 38,  3,  6,  3,  0,  0,
1020        26, 12, 10,  4,  4,  0,  1,  1,  3,  0,  0}
1030 m_vset(83,MARIMBA)
1040 dim char TRUMPET(4,10)={
1050        58, 15,  0,  0,  0,  0,  0,  0,  0,  3,  0,
1060        10, 14,  0,  4,  1, 24,  2,  2,  3,  0,  0,
1070        11, 14,  0, 10, 15, 38,  2, 14,  0,  0,  0,
1080        10, 14,  0,  4,  1, 38,  2,  2,  7,  0,  0,
1090        19, 15,  0,  8,  0,  0,  1,  2,  0,  0,  0}
1100 m_vset(84,TRUMPET)
1110 /*
1120 /*    初期設定
1130 /*
1140 m_init()
1150 for i=1 to 8:m_alloc(i,4096):m_assign(i,i):next
1160 str a(10)[256]
1170 m_tempo(130)
1180 /*
1190 /*    M M L   &   m _ t r k ( )
1200 /*
1210 a(0)="v9q818o4@77
1220 a(1)="|:d4.dr2r1r1r2..|1r:||2
1230 a(2)="v10o3@84 >b|:<ba+ba+ba+beg4b4r4.>b<agaga4eg4.r2r>b:|
1240 a(3)="|:<ag+ag+ag+ag+a4b4r4.>b<agaga4eg4.r2|1r>b:||2<eg
1250 a(4)="|:r2.v9o3@77b&{b-&a&a-}&g2r4v10o3@84egr2.v9o3@77|1<c
&{>b&b-&a&a-&}g2r4v10o3@84eg:||2f&{f+&g&g+&a&a+&b}&<c2r4
1260 a(5)="rv9o5@83 >{brra}4{b<degab}4<d>b{aba}{gf+} {edre}4{de
d}{>ba}{gf+gbagf+e}2 ed{ded}{>bagr}4{rg}{b<de>b}4 {<def+grgra}2{
gabga<d}4{degb}4
1270 a(6)="b{ag}{aggf+aagf+}2{aa-gf+ bba<d}2{d>b}<ed{cdc}>ba {<
cdc>b<c>babagagfgfefededcdc}1 {>b<c>babagagfgf}2{efeded}4cc
1280 a(7)="{<c+>brf+<c+f+c+>b+<f+<c+>f+e+<c+rc+>g}1 {f+<c+>ageb
edc+f+ec+>b<ee-d}1 {>brb<c}4{>b<degab}4<d>b{bbagf+e}4 {f+edf+edf
+e}2{df+ed}4 {f+edc+}{>b}
1290 a(8)="<{f+ed+c+f+ed+c+}2{f+ed+c+d+ef+ed+c+d+e}2 {f+ed+c+d+
ef+ed+c+>ba}2bb{bag+}f+ {rrrdededegegaga<c>a<cedegeg}1 {aga<c>a<
cdcdede}2{gegaga}4<d>b <c
1300 a(9)="v10o3@77 rb64<c4..&c32.>b4 {g<c>bg}4{rg} g64a16..&a4
r2..{efefrgrf}2e{c>agfedce}2{dc}>ag2.r4<<{rar<drerd}2{r>a}g{rc}>
ar4.<a{r<e-dc}4d>a<c4d16c{grgf}4r16e
1310 for i=0 to 4:m_trk(1,a(i)):next
1320 m_trk(1,"r")
1330 for i=2 to 4:m_trk(1,a(i)):next
1340 m_trk(1,"r")
1350 for i=5 to 9:m_trk(1,a(i)):next
1360 for i=2 to 3:m_trk(1,a(i)):next
1370 for i=1 to 3:m_trk(1,a(4)+"v10o3@84eg"):next
1380 m_trk(1,a(4))
1390 /*
1400 /*
1410 a(0)="v4q818o3@80 |:16r4ee:|
1420 a(1)="v4o3@80 |:16r4ee:|
1430 a(2)="v6o2@79 e1f+2e2d1&d1e1f+2e2d2.>g4a4<c4e4g4
1440 a(3)="v7|:g1&g1&g1&g1:|
1450 for i=0 to 3:m_trk(2,a(i)):next
1460 for i=1 to 3:m_trk(2,a(i)):next
1470 for i=1 to 3:m_trk(2,a(i)):next
1480 for i=1 to 2:m_trk(2,a(i)):next
1490 for i=1 to 4:m_trk(2,a(3)):next
1500 /*
1510 /*
1520 a(0)="v6q818o3@82 |:d1r1r1d1:|
1530 a(1)="|:d1r1c1d1:|
1540 a(2)="v6o2@79 c+1&c+1>a1b1<c+1&c+1>a2.r4<e2e2
1550 a(3)="v6o3@82 >e1<e1d1g1>b1<e1d1
1560 for i=0 to 3:m_trk(3,a(i)):next
1570 m_trk(3,"g1")
1580 for i=1 to 3:m_trk(3,a(i)):next
1590 m_trk(3,"e1")
1600 for i=1 to 3:m_trk(3,a(i)):next
1610 m_trk(3,"e1")
1620 for i=1 to 2:m_trk(3,a(i)):next
1630 for i=1 to 4:m_trk(3,a(3)+"g1"):next
1640 /*
1650 /*
1660 a(0)="v6q818o2@78
1670 a(1)=">g1b2<d2c2.>b4g1g1b2<d2e2.d4>b1
1680 a(2)="v6o1@78 |:g1b2.<d4c2.>b4g1:|
1690 a(3)="<c+1&c+1d1>b1<c+1&c+1r1r1
1700 a(4)="v6o3@81p1 |:8r{gagf}4e{gagf}4er:|
1710 for i=0 to 4:m_trk(4,a(i)):next
1720 for i=2 to 4:m_trk(4,a(i)):next
1730 for i=2 to 4:m_trk(4,a(i)):next
1740 for i=2 to 3:m_trk(4,a(i)):next
1750 for i=1 to 4:m_trk(4,a(4)):next
1760 /*
1770 /*
1780 a(1)=">e1g2b2a2.g4e1e1g2b2<c2.>b4g1
1790 a(2)="v6o1@78 |:e1g2.b4a2.g4e1:|
1800 a(3)="a1&a1a1g1a1&a1r1r1
1810 a(4)="v6o3@81p2 |:8r{>b<c>ba}4g{b<c>ba}4g<r:|
1820 for i=0 to 4:m_trk(5,a(i)):next
1830 for i=2 to 4:m_trk(5,a(i)):next
1840 for i=2 to 4:m_trk(5,a(i)):next
1850 for i=2 to 3:m_trk(5,a(i)):next
1860 for i=1 to 4:m_trk(5,a(4)):next
1870 /*
1880 /*
1890 a(0)="v11q818o3@76
1900 a(1)="|:e4.e>bb4b<e4.ef+ga+bc4.c>gg4g<c4.cd+ef+g:|
1910 a(2)="|:e4.e>bb4b<e4.e>bb4b<c4.c>gg4g<c4.c>|1ab<dd+:||2>gg
aa+
1920 a(3)="|:b4.bf+f+4f+b4.bf+f+af+<|1c4.c>gg4g<c4.c>ggaa+:||2<
g4.gddgd>g4g4<efga
1930 a(4)="|:c4.c>gg4g<c4.c{ce}>eff+<g4.gddgdg4.gaged:|
1940 for i=0 to 4:m_trk(6,a(i)):next
1950 for i=2 to 4:m_trk(6,a(i)):next
1960 for i=2 to 4:m_trk(6,a(i)):next
1970 for i=2 to 3:m_trk(6,a(i)):next
1980 for i=1 to 4:m_trk(6,a(4)):next
1990 /*
2000 /*
2010 a(0)="q818o4
2020 a(1)="@75>v7|:7gggg:|rgrg|:7gggg:|r2<
2030 a(2)="@75v15<g4.>@73v13g|:14v15gv13gv14gv13g:|v15gv13gv14g
@74g
2040 a(3)="@73|:15v15gv13gv14gv13g:|r2
2050 a(4)="@75v15<g4.v7g{gggg}4gg|:7g{gggg}4g{gggg}4gg:|>
2060 a(5)="<|:24g{gggg}4g{gggg}4gg:|>
2070 for i=0 to 4:m_trk(7,a(i)):next
2080 for i=2 to 4:m_trk(7,a(i)):next
2090 for i=2 to 4:m_trk(7,a(i)):next
2100 for i=2 to 5:m_trk(7,a(i)):next
2110 /*
2120 /*
2130 a(0)="v15q818o3
2140 a(1)="|:3 @72cr@71br@72cc@71b@72c:|cr@71brb@72c@71b@72c|:3
cr@71br@72cc@71b@72c:|cr@71brbbbb
2150 a(2)="|:15@72cr@71br@72cc@71b@72c:|cr@71brbbbb
2160 a(3)="|:3 @72cr@71br@72cc@71b@72c:|cr@71brb@72c@71b@72c|:3
cr@71br@72cc@71b@72c:|cr@71brbbbb
2170 a(4)="|:15@72cr@71br@72cc@71b@72c:|cr@71b@72cc@71bbb
2180 a(5)="|:7 @72cr@71br@72cc@71b@72c:|cr@71brbbbb
2190 a(6)="|:15@72cr@71br@72cc@71b@72c:|cr@71br{bb}b@72v13{gg}g
v15
2200 a(7)="|:  @72cr@71br@72cc@71b@72c |1cr@71brb@72c@71b@72c:|
|2cr@71brbbbb
2210 for i=0 to 3:m_trk(8,a(i)):next
2220 for i=2 to 6:m_trk(8,a(i)):next
2230 m_trk(8,a(3))
2240 m_trk(8,a(2))
2250 m_trk(8,a(5))
2260 /*
2270 /*    P l a y i n g
2280 /*
2290 m_play()
```

リスト5 ソーサリアン

```
 10 /*/////////////////////////////////////////////////////////////
 20 /*
 30 /*
 40 /*           そ～さりあんあんあんあん
 50 /*
 60 /*
 70 /*           あかけ゛のあん
 80 /*
 90 /*           し゛ゃなくて　せいかん　のて～ま
100 /*
110 /*                  Programmed  by  北野  直之
120 /*
```

▶胃がいったいー，胃がいったいー，明日からテストだ，胃がいったいー。喜んでいるようだが，本気で胃が痛い田村。　　田村　憲生 (20) 広島県

```
130 /*                              Copyright 日本ファルコム
140 /*
150 /*////////////////////////////////////////////////////////////
160 /*
170 /* 音色設定
180 /*
190 /*------------------------------------ シンセサウンド 1
200 /*
210 /*   アルゴリズム４番によって厚みのある音を出しています。
220 /*
230 /*   マルチプルの値を大きく取ったところがポイント。
240 /*
250 dim char v(4,10)={
260 /*    AF   OM   WF   SY   SP  PMD  AMD  PMS  AMS  PAN
270       60,  15,   0,   0,   0,   0,   0,   0,   0,   3,   0,
280 /*    AR   DR   SR   RR   SL   OL   KS   ML  DT1  DT2  AME
290       31,   0,   0,   0,   0,  33,   0,   8,   3,   0,   0,
300       20,  15,   8,   7,   3,   0,   0,   8,   3,   0,   0,
310       31,   0,   0,   0,   0,  23,   0,   4,   7,   0,   0,
320       31,  15,   8,   7,   3,   0,   0,   4,   7,   0,   0}
330 m_vset(1,v)
340 /*
350 /*---------------------------------------- シンセベース
360 /*
370 /*   これもアルゴリズム４番による音色。
380 /*
390 /*   デチューン１によって「うねり」を出しています。
400 /*
410 v={
420 /*    AF   OM   WF   SY   SP  PMD  AMD  PMS  AMS  PAN
430       52,  15,   0,   0,   0,   0,   0,   0,   0,   3,   0,
440 /*    AR   DR   SR   RR   SL   OL   KS   ML  DT1  DT2  AME
450       31,   0,   0,   0,   0,  23,   0,   2,   3,   0,   0,
460       20,   5,   4,   6,   5,   4,   0,   8,   3,   0,   0,
470       31,   0,   0,   0,   0,  23,   0,   4,   7,   0,   0,
480       31,   5,   4,   6,   5,   4,   0,   8,   7,   0,   0}
490 m_vset(2,v)
500 /*
510 /*------------------------------------------ ストリングス
520 /*
530 /*   キャリアのアタックレートをかなり小さく取って
540 /*
550 /*   なめらかなストリングスらしさを出しています。
560 /*
570 v={
580 /*    AF   OM   WF   SY   SP  PMD  AMD  PMS  AMS  PAN
590       59,  15,   0,   0,   0,   0,   0,   0,   0,   3,   0,
600 /*    AR   DR   SR   RR   SL   OL   KS   ML  DT1  DT2  AME
610       31,   0,   0,   5,   0,  30,   1,   4,   3,   0,   0,
620       31,   0,   0,   5,   0,  30,   1,   8,   0,   0,   0,
630       31,   0,   0,   5,   0,  40,   1,  12,   7,   0,   0,
640       10,   0,   0,   6,   0,   0,   1,   8,   0,   0,   0}
650 m_vset(3,v)
660 /*
670 /*------------------------------------------ ＰＳＧベース
680 /*
690 /*   この音色はまさにＰＳＧ丸出し。
700 /*
710 /*   マルチプルを２：１に設定して再現しました。
720 /*
730 v={
740 /*    AF   OM   WF   SY   SP  PMD  AMD  PMS  AMS  PAN
750       61,  15,   0,   0,   0,   0,   0,   0,   0,   3,   0,
760 /*    AR   DR   SR   RR   SL   OL   KS   ML  DT1  DT2  AME
770       31,   0,   0,   0,   0,  24,   0,   2,   0,   0,   0,
780       31,  15,   0,   7,   3,   6,   0,   1,   0,   0,   0,
790       31,  15,   0,   8,   3,   6,   0,   1,   0,   0,   0,
800       31,  15,   0,   8,   3,   6,   0,   1,   0,   0,   0}
810 m_vset(4,v)
820 /*
830 /*-------------------------------------------- バスドラム
840 /*
850 /*   オペレータ１、２でノイズを、
860 /*
870 /*   ３、４でトーンを出して作ってみました。
880 /*
890 v={
900 /*    AF   OM   WF   SY   SP  PMD  AMD  PMS  AMS  PAN
910       60,  15,   0,   0,   0,   0,   0,   0,   0,   3,   0,
920 /*    AR   DR   SR   RR   SL   OL   KS   ML  DT1  DT2  AME
930       31,   0,  21,   0,   0,  10,   0,   3,   0,   1,   0,
940       31,   0,  20,  15,   0,   2,   0,   3,   0,   0,   0,
950       31,   0,  18,  15,   0,  20,   0,   1,   0,   0,   0,
960       31,   0,  18,  15,   0,   0,   0,   1,   0,   0,   0}
970 m_vset(5,v)
980 /*
990 /*--------------------------------------------- ハイハット
1000 /*
1010 /*   一応アルゴリズム４番で作っていますが、
1020 /*
1030 /*   基本的にはノイズだけなのでどれでも同じだと思います。
1040 /*
1050 v={
1060 /*    AF   OM   WF   SY   SP  PMD  AMD  PMS  AMS  PAN
1070      60,  15,   0,   0,   0,   0,   0,   0,   0,   3,   0,
1080 /*    AR   DR   SR   RR   SL   OL   KS   ML  DT1  DT2  AME
1090      31,  31,   5,   5,   2,   0,   0,  15,   0,   3,   0,
1100      31,  19,  16,   8,  10,   8,   0,   9,   0,   2,   0,
1110      31,  11,   6,   6,  10,   0,   0,  15,   0,   2,   0,
1120      31,  22,  17,   8,  10,   8,   0,  15,   0,   1,   0}
1130 m_vset(6,v)
1140 /*
1150 /*------------------------------------------ スネアドラム
1160 /*
1170 /*   この音色はバスドラムのパラメータを
1180 /*
1190 /*   エディットして作りました。
1200 /*
1210 v={
1220 /*    AF   OM   WF   SY   SP  PMD  AMD  PMS  AMS  PAN
1230      60,  15,   0,   0,   0,   0,   0,   0,   0,   3,   0,
1240 /*    AR   DR   SR   RR   SL   OL   KS   ML  DT1  DT2  AME
1250      31,   0,   0,   0,   0,   0,   0,  15,   0,   0,   0,
1260      31,  15,  15,   7,   4,   4,   0,   0,   0,   0,   0,
1270      31,  22,  21,  15,  10,   0,   0,   4,   0,   0,   0,
1280      31,  18,  17,  15,  10,   0,   0,   1,   0,   0,   0}
1290 m_vset(7,v)
1300 /*
1310 /*---------------------------------------- トランペット
1320 /*
1330 /*   ８８ＳＲ内蔵音色２１番です。
1340 /*
1350 /*   よく似ていたので引っ張ってきました。
1360 /*
1370 v={
1380 /*    AF   OM   WF   SY   SP  PMD  AMD  PMS  AMS  PAN
1390      58,  15,   0,   0,   0,   0,   0,   0,   0,   3,   0,
1400 /*    AR   DR   SR   RR   SL   OL   KS   ML  DT1  DT2  AME
1410      13,   6,   2,   8,   1,  25,   2,   4,   3,   0,   0,
1420      15,   8,   0,   8,   1,  32,   1,  12,   5,   0,   0,
1430      21,   7,   0,   8,   2,  45,   0,   4,   3,   0,   0,
1440      18,   4,   0,   8,   2,   0,   1,   4,   4,   0,   0}
1450 m_vset(8,v)
1460 /*
1470 /*------------------------------------------- ピアノ
1480 /*
1490 /*   ８８ＳＲシリーズの内蔵音色１３番です。
1500 /*
1510 /*   なかなかいい音ですよ！
1520 /*
1530 v={
1540 /*    AF   OM   WF   SY   SP  PMD  AMD  PMS  AMS  PAN
1550      60,  15,   0,   0,   0,   0,   0,   0,   0,   3,   0,
1560 /*    AR   DR   SR   RR   SL   OL   KS   ML  DT1  DT2  AME
1570      31,   0,   8,   0,   4,  32,   3,   4,   2,   0,   0,
1580      25,   7,   7,   7,   3,   2,   2,   4,   0,   0,   0,
1590      31,   0,   8,   0,   4,  28,   3,  12,   5,   0,   0,
1600      25,   8,   6,   7,   3,   0,   2,   4,   1,   0,   0}
1610 m_vset(9,v)
1620 /*
1630 /*-------------------------------------- グラスベル
1640 /*
1650 /*   澄んだ響きを出すためにアルゴリズム４番で、
1660 /*
1670 /*   うまくデチューンをかけることができました。
1680 /*
1690 v={
1700 /*    AF   OM   WF   SY   SP  PMD  AMD  PMS  AMS  PAN
1710      60,  15,   0,   0,   0,   0,   0,   0,   0,   3,   0,
1720 /*    AR   DR   SR   RR   SL   OL   KS   ML  DT1  DT2  AME
1730      31,   0,   4,   3,   0,  35,   1,   4,   3,   0,   0,
1740      31,   0,   7,   7,   0,   0,   2,   1,   3,   0,   0,
1750      31,   0,   7,   3,   0,  40,   1,  15,   7,   0,   0,
1760      31,   0,   7,   7,   0,   0,   1,   4,   7,   0,   0}
1770 m_vset(10,v)
1780 /*
1790 /* 初期設定
1800 /*
1810 m_init():for i=1 to 8:m_alloc(i,3000):next
1820 for i=1 to 8:m_assign(i,i):next
1830 str a[256],b[256],c[256],d[256],e[256],f[256],g[256],h[256
],j[256],k[256],l[256],m[256]
1840 str n[256],o[256],p[256],q[256],r[256],s[256],t[256],u[256
],w[256],x[256],y[256],z[256]
1850 /*
1860 /* 演奏データ
1870 /*
1880 a="t128  |:256 o3 l16 @1 p3 q8   v13 y48,12"
1890 b="e>bg+< e>bg+< ec+>a< ec+>a< f+d>a< f+d>a< g+e>b< g+e>b<
g+e>b< g+e>b<
1900 c="f+c+>a< f+c+>a< g+e>b< g+e>b< af+c+ af+c+ af+d af+d g+e
>b< g+e>b<
1910 d="f+d>a< f+d>a< af+d af+d
1920 e="ae>b< ae>b< ae>b< ae>b< g+e>b< g+e>b< g+e>b< g+e>b<
1930 f="@8o2v14p1rrraaaa4. rrraaaa4. rrraaaa4. rrraaaaaaaa rrr
aaaa4. rrraaaa4. rrraaaa4.
1940 g="<e8.e8.eeeeee>
1950 h="<f+8.e8.d8.e8.
1960 j="@10v13o4a2.a4.a8b8<c8>b4.g2.<d4.e2.e4.e8d8c8d8e8d8>g2.g
4.
1970 k="a2.l8agfab<c>b4.<g2.>b4.<c2.cdedefge>b<e>bge4.b<cd
1980 l=">a2.a-2.g4.b4.a4.b8.<c+8.d2.d4.>f8.a8.a4.g4.a4.b4.
1990 m="b2.b4.b4.a4.<c4.e4.e8.d8.>a2.a2.b2.b2.
2000 z=":|"
2010 /*
2020 m_trk(1,a)
2030 m_trk(1,b)
2040 m_trk(1,c)
2050 m_trk(1,d)
2060 m_trk(1,e+e)
2070 m_trk(1,f)
2080 m_trk(1,g)
2090 m_trk(1,f)
2100 m_trk(1,h)
2110 m_trk(1,j)
2120 m_trk(1,k)
2130 m_trk(1,l)
2140 m_trk(1,m)
2150 m_trk(1,z)
2160 /*
```

▶共通一次が終わった瞬間，私は思った。「ああ，こうやって浪人は生まれるんだナ」。来年はガンバル。

北條　育男 (18) 埼玉県

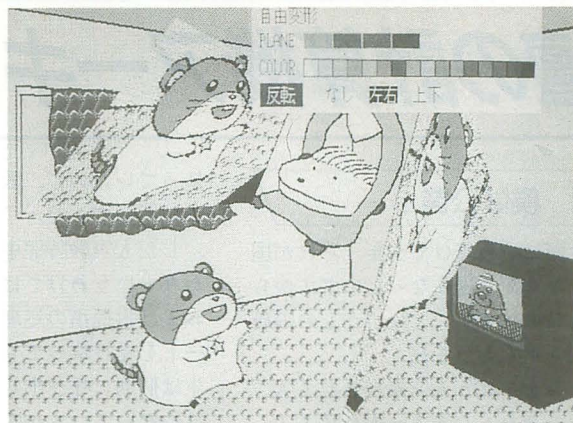
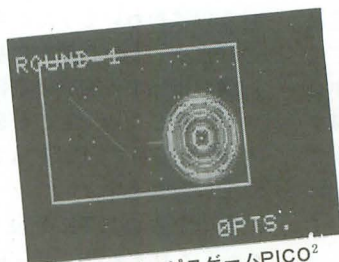
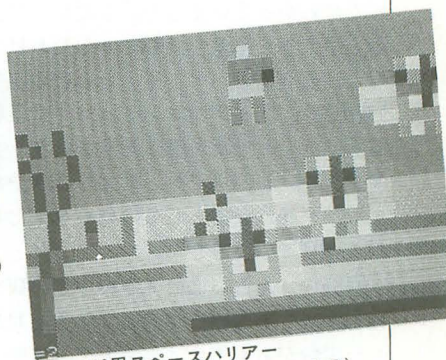
```
 2170 /*
 2180 a="       |:256 o2 l16 @1 p3 q8   v13 y49,20"
 2190 b="bg+e bg+e <c+>ae <c+>ae <d>af+ <d>af+ <e>bg+ <e>bg+ <e>
bg+ <e>bg+
 2200 c="<c+>af+ <c+>af+ <e>bg+ <e>bg+< f+c+>a< f+c+>a< f+d>a< f
+d>a< e>bg+< e>bg+<
 2210 d="d>af+< d>af+< f+d>a< f+d>a<
 2220 e="e>ba< e>ba< e>ba< e>ba< e>bg+< e>bg+< e>bg+< e>bg+<
 2230 f="@9o2c+4>a<c+e4c+ea4.<c+8>b8a8g2.g4.g8a8g8f+df+a8.af+a<d
4.&d8.>a8b8a8a4.a4.ag+eag+eag+ebg+e
 2240 g="c+4>a<c+e4c+ea4.<c+8>b8a8g2.g4.g8a8g8f+df+a8.af+a<d4.&d
8.>a8b8a8a4.a4.ag+eag+eag+eag+e
 2250 h="@10v13o4f2.f4.f8g8a8g4.e2.b4.<c2.c4.c8>b8a8b8<c8>b8e2.e
4.
 2260 j="f2.l8fedfgag4.<e2.>g4.a2.ab<c>b<cde>bgbge>b4.<gab
 2270 k="l8f2.f2.e4.e4.<a8.g8.f8.e8.>a2.a4.d8.f8.f4.f4.f4.g4.
 2280 l="g+2.g+4.e4.e4.a4.<c4.c8.>b8.f2.f2.g2.g+2.
 2290 m_trk(2,a)
 2300 m_trk(2,b)
 2310 m_trk(2,c)
 2320 m_trk(2,d)
 2330 m_trk(2,e+e)
 2340 m_trk(2,f)
 2350 m_trk(2,g)
 2360 m_trk(2,h)
 2370 m_trk(2,j)
 2380 m_trk(2,k)
 2390 m_trk(2,l)
 2400 m_trk(2,z)
 2410 /*
 2420 /*
 2430 a="       |:256 o3 l16 @1 p3 q8   v13 y50,28"
 2440 b="g+e>b< g+e>b< aec+ aec+ af+d af+d bg+e bg+e bg+e bg+e
 2450 c="af+c+ af+c+ bg+e bg+e <c+>af+ <c+>af+
 2460 d="<d>af+ <d>af+ bg+e bg+e af+d af+d <d>af+ <d>af+
 2470 e="bae bae bae bae bg+e bg+e bg+e bg+e
 2480 f="@8o2v14p2rrraaaa4. rrraaaa4. rrraaaa4. rrraaaaaaaaa rrr
aaaa4. rrraaaa4. rrraaaa4.
 2490 g="<e8.e8.eeeeee>
 2500 h="<f+8.e8.d8.e8.
 2510 j="@10v9p2o4r8a2.a4.a8b8<c8>b4. g2.<d4.e2.e4.e8d8c8d8e8d8>
g2.g4.
 2520 k="a2.l8agfab<c>b4.<g2.>b4.<c2.cdedefge>b<e>bge4.b<cd
 2530 l=">a2.a-2.g4.b4.a4.b8.<c+8.d2.d4.>f8.a8.a4.g4.a4.b4.
 2540 m="b2.b4.b4.a4.<c4.e4.e8.d8.>a2.a2.b2.b2&b8
 2550 /*
 2560 m_trk(3,a)
 2570 m_trk(3,b)
 2580 m_trk(3,c)
 2590 m_trk(3,d)
 2600 m_trk(3,e+e)
 2610 m_trk(3,f)
 2620 m_trk(3,g)
 2630 m_trk(3,f)
 2640 m_trk(3,h)
 2650 m_trk(3,j)
 2660 m_trk(3,k)
 2670 m_trk(3,l)
 2680 m_trk(3,m)
 2690 m_trk(3,z)
 2700 /*
 2710 /*
 2720 a="       |:256 o1 l16 @2 p3 q8   v14 y51,16"
 2730 b="a2.&a4.&a4<c+ef+2.&f+4.&f+4ec+ d2.&d4.d8e8f+8
 2740 c="e1&e2 o2 @3 q7 v14 eab<e>ba eab<e>ba eg+b<e>bg+ eg+b<e>
bg+
 2750 d="e2.<e4.e8d8c+8>e2.a4.<e4.e4.d2.d8e8d8e4.d4.c+8.>b8.a8.b
8.
 2760 e="e2.<e4.e8d8c+8e4.>a4.<e4.a4.a4.d2.a8g+8f+8a4.d4.a8.g+8.
f+8.g+8.
 2770 f="v13q7> d>a<dfdfafdfd>a< d>a<dfdfafdfd>a< e>b<egegbgege>
b<e>b<egega<bc>b<cd
 2780 g="ec>a<":g=g+g+g+g+g+g+g+g:h="d>bg<":h=h+h+h+h+h+h+h+h:g=
g+h
 2790 h="> d>a<dfdfafdfd>a< d>a<dfdfafdfd>a< e>b<egegbgege>b<e>b
<egega<bc>b<cd
 2800 j="ec>a<":j=j+j+j+j+j+j+j+j:j=j+"e>bg< e>bg< e>bg< e>bg e4
.<v14e8f8g8
 2810 k="l8v15a4.g.f.a-4.a-b-a-g4.b.<d.c+4.d.e.f2.f4.>a.<c.c4.>b
4.<c4.d4.
 2820 l="e2.l8efedc>b<d4.&d8.e8.c4.&c8.l16c>ba<c2.c4.c8>b8<c8d2.
d2.
 2830 /*
 2840 m_trk(4,a)
 2850 m_trk(4,b)
 2860 m_trk(4,c)
 2870 m_trk(4,d)
 2880 m_trk(4,e)
 2890 m_trk(4,f)
 2900 m_trk(4,g)
 2910 m_trk(4,h)
 2920 m_trk(4,j)
 2930 m_trk(4,k)
 2940 m_trk(4,l)
 2950 m_trk(4,z)
 2960 /*
 2970 /*
 2980 a="       |:256 o2 l16 @4 p3 q8   v12 y52,20"
 2990 c="e8r e8r e8r e8r e8r e8r e8r e8r
 3000 d="a8ra8ra8ra8ra8ra8ra8ra8rg8rg8rg8rg8rg8rg8rg8rg8rf+8rf+8
rf+8rf+8rf+8rf+8rf+8rf+8r
 3010 e="f8rf8rf8rf8re8re8r>ee<e>ee<e
 3020 f="f8rf8rf8rf8rf8rf8rf8rf8re8re8re8re8re8re8re8re8r":f=f+f
+f
 3030 g="f8rf8rf8rf8rf8rf8rf8rf8re8re8re8re8re4.e8e8e8
 3040 h="f8rf8rf8rf8rf8rf8rf8rf8re8re8re8re8ra8ra8ra8ra8r
 3050 j="d8rd8rd8rd8rd8rd8rd8rd8rg8rg8rg8rg8r>g8rgggggggg
 3060 k="<e8re8re8re8re8re8re8re8ra8ra8ra8ra8ra8ra8ra8ra8rf8rf8r
f8rf8rf8rf8rf8rf8rg8rg8rg8rg8re2.
 3070 /*
 3080 m_trk(5,a)
 3090 m_trk(5,b)
 3100 m_trk(5,c+c)
 3110 m_trk(5,d)
 3120 m_trk(5,e)
 3130 m_trk(5,d)
 3140 m_trk(5,e)
 3150 m_trk(5,f)
 3160 m_trk(5,g)
 3170 m_trk(5,h)
 3180 m_trk(5,j)
 3190 m_trk(5,k)
 3200 m_trk(5,z)
 3210 /*
 3220 /*
 3230 a="       |:256 o1 l16 @4 p3 q8   v13 y53,22"
 3240 /*
 3250 m_trk(6,a)
 3260 m_trk(6,b)
 3270 m_trk(6,c+c)
 3280 m_trk(6,d)
 3290 m_trk(6,e)
 3300 m_trk(6,d)
 3310 m_trk(6,e)
 3320 m_trk(6,f)
 3330 m_trk(6,g)
 3340 m_trk(6,h)
 3350 m_trk(6,j)
 3360 m_trk(6,k)
 3370 m_trk(6,z)
 3380 /*
 3390 /*
 3400 a="       |:256 o3 l16 @1 p1 q8   v13 y54,36"
 3410 b="e>bg+< e>bg+< ec+>a< ec+>a< f+d>a< f+d>a< g+e>b< g+e>b<
 g+e>b< g+e>b<
 3420 c="f+c+>a< f+c+>a< g+e>b< g+e>b< af+c+ af+c+ af+d af+d g+e
>b< g+e>b<
 3430 d="f+d>a< f+d>a< af+d af+d
 3440 e="ae>b< ae>b< ae>b< ae>b< g+e>b< g+e>b< g+e>b< g+e>b<
 3450 f="o2 @3 v10 q7 r8.eab<e>ba eab<e>ba eg+b<e>bg+ eg+b<e>bg+
 v11
 3460 g="e2.<e4.e8d8c+8>e2.a4.<e4.e4.d2.d8e8d8e4.d4.c+8.>b8.a8.b
8.
 3470 h="e2.<e4.e8d8c+8e4.>a4.<e4.a4.a4.d2.a8g+8f+8a4.d4.a8.g+8.
f+8.g+16.
 3480 j="v9q6> d>a<dfdfafdfd>a< d>a<dfdfafdfd>a< e>b<egegbgege>b
<e>b<egega<bc>b<cd
 3490 k="ec>a<":k=k+k+k+k+k+k+k+k:l="d>bg<":l=l+l+l+l+l+l+l+l:k=
k+l
 3500 l=">d>a<dfdfafdfd>a< d>a<dfdfafdfd>a< e>b<egegbgege>b<e>b<
egega<bc>b<cd
 3510 m="ec>a<":m=m+m+m+m+m+m+m+m:m=m+"e>bg< e>bg< e>bg< e>bg r1
6.e4.<v11e8f8g8
 3520 n="l8v12a4.g.f.a-4.a-b-a-g4.b.<d.c+4.d.e.f2.f4.>a.<c.c4.>b
4.<c4.d4.
 3530 o="e2.l8efedc>b<d4.&d8.e8.c4.&c8.l16c>ba<c2.c4.c8>b8<c8d2.
d2&d16
 3540 /*
 3550 m_trk(7,a)
 3560 m_trk(7,b)
 3570 m_trk(7,c)
 3580 m_trk(7,d)
 3590 m_trk(7,e)
 3600 m_trk(7,f)
 3610 m_trk(7,g)
 3620 m_trk(7,h)
 3630 m_trk(7,j)
 3640 m_trk(7,k)
 3650 m_trk(7,l)
 3660 m_trk(7,m)
 3670 m_trk(7,n)
 3680 m_trk(7,o)
 3690 m_trk(7,z)
 3700 /*
 3710 /*
 3720 a="       |:256 o1 l16 @5 p3 q8 @v127 y55,20"
 3730 b="cr2rcc cr2rcc cr2rcc cr2rcc cr2rcc c4.c8c8c8
 3740 c="l8c.c.c.c.c.c.c.c.@5c.@7c.@5c.@7c.@5c.@7c.l16@5c@7ccccc
l8
 3750 d="@5c.@7c.@5c.@7c.@5c.@7c.@5c.@7c.@5c.@7c.@5c.@7c.@5c.@7c
.@5c.@7c.
 3760 e="l16@5c8@6c@7c8@6c@5c8@6c@7c8@6c":e=e+e:e=e+e:e=e+e
 3770 g="l16@5c8@6c@7c8@6c@5c8@6c@7c8@6c":f=g+g:f=f+f:f=f+g+g+g:
f=f+"@5c8@6c@5c8@6c@7c@6c@7c@6c@7c@6c
 3780 j="l16@5c8@6c@7c8@6c@5c8@6c@7c8@6c":h=j+j:h=h+h:h=j+j+j+j+
j+j+j+"@5c8@6c@7c8@6c@5c@7cc@5c@7cc
 3790 k=j+j:k=k+k+j+j+j+"@5c8.c8.c8.c8.
 3800 /*
 3810 m_trk(8,a)
 3820 m_trk(8,b)
 3830 m_trk(8,c)
 3840 m_trk(8,d+d)
 3850 m_trk(8,d+d)
 3860 m_trk(8,e)
 3870 m_trk(8,f)
 3880 m_trk(8,h)
 3890 m_trk(8,k)
 3900 m_trk(8,z)
 3910 m_play()
 3920 end
```

投稿プログラム大募集

のお知らせ

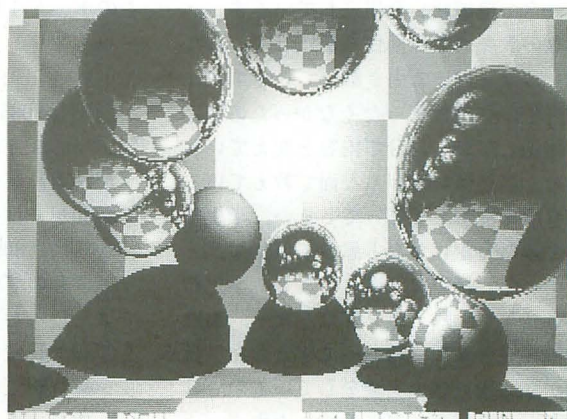
Oh!Xでは，毎月さまざまな投稿プログラムを掲載しております。これらはすべて，ゲーム音楽を聞いているうちに自分のマシンで演奏してみたくなった，市販のものもあるけどもっと便利なグラフィックツールが欲しかった，またはMZ-700でスペースハリアーを遊びたいなど，どれも皆さんが日常のなかでパソコンと接しているうちに，ふと思いついたことを形にしようと努力して生み出された傑作，名作ばかりなのです。

でも，読者の皆さんがそうして作り上げたプログラムを，一部の方を除いては自分のディスクのなかだけにしまっておくのはもったいない話。ひとりでも多くのユーザーに使ってもらえば，またそれをベースにして新しいプログラムが生まれる可能性だって広がるのです。

ですから，Oh!Xではそういったちょっとしたきっかけを機に，完成度の高いものよりも自分のアイデアをそのまま形にしたような，オリジナリティあふれる投稿プログラムをスペースを空けてお待ちしています。もちろん，ピコピコゲームのようなショートプログラムも大歓迎。自信作をお持ちの方は，募集要項をよくお読みのうえぜひご参加ください。お待ちしています。

MZ-2500用グラフィックツールDMACS（1988年9月号）

MZ-2500用ピコピコゲームPICO²（1988年4月号）

MZ-700用スペースハリアー（1988年10月号）

X1/X1 turbo用割り込みミュージックシステムPSI（1988年3月号）

X68000用ストラテジーゲームSTAR TREK（1988年11月号）

S-OS"SWORD"用ELFES（1988年2月号）

X1turbo用レイトレーシングツールturbo RAY TRACER（1988年9月号）

投稿募集要項

1) お送りいただくプログラムには，住所・氏名・年齢・職業・連絡先電話番号・機種名・使用言語・必要な周辺機器・マイコン歴等を明記のうえ，封書の宛て先の最後には「Oh!X LIVE」や「S-OS"SWORD"」，「投稿ゲームプログラム」など，プログラムの内容を明確にご記入ください。

2) 投稿されるプログラムには，詳しい内容を記入した原稿と一緒にフローチャート，変数表，メモリマップ，参考文献などの資料もお書き添えのうえお送りください。また，お送りいただいた原稿については，当方で加筆，修正させていただく場合があります。

3) お送りいただくプログラムは最低2回はセーブしてください。基本的に同封されたカセットテープおよびフロッピーディスクについてはご返送いたしませんので，あらかじめご了承ください。

4) ハード製作関係の投稿につきましては，最初は詳しい内容のわかる原稿のみお送りいただければ結構です。その後，当方において製作物が必要だと判断した場合は，改めてご連絡いたします。

5) お送りいただいた投稿プログラムの採用につきましては，掲載月号が決定した時点で当方よりご連絡を差し上げます。特に各種ツール関係，ハード関係のものにつきましては，特集内容などを考慮したうえで採用が決定されることがありますので，採用結果をご連絡するまでに時間がかかってしまう場合もあります。

6) 投稿いただいたプログラムにバグ等が発見された場合には，新しいプログラムの入ったメディアと一緒に，文書にてご連絡ください。

7) 掲載された投稿プログラムに対しては当社規定の原稿料をお支払いいたします。また，プログラムの著作権等は制作された方に保留されますが，PDSとしてネットなどにアップロードされる場合は，必ず編集室まで事前にご連絡ください。なお，一般的モラルとして，他誌との二重投稿または，他誌に掲載されたプログラムの移植などについては固くお断わりいたします。

宛て先
〒102 東京都千代田区九段南2-3-26 井関ビル
日本ソフトバンク Oh!X編集室「投稿プログラム」係

教壇の計算機アーキテクト

僕は公僕！

授業料を国に納め続けて9年，それが国からお金をもらう立場になったのですからこれは大きな違いです。とはいうものの研究室に出入りする人たちを，

学生＝お金を払う人＝奉仕される人
教員＝お金をもらう人＝奉仕する人

のように明快に分けられることは僕にもとっくにわかっていました。したがって意識の上で大きな落差を感じたりしたわけではなく，研究室のバイト王といわれる院生に「有田さん，アップルコンピュータに就職するとMac SEが個人用に支給されるそうですよ」(本当らしい!?)と耳元でささやかれても，平静に公僕としての務めを果たしている今日このごろです。

自分の，あるいは研究室の中での研究以外にやることには，学部の2年生を対象とした実験があります。電気系工学実験というもので，テーマのひとつ「マイクロコンピュータと機械語」を担当しています。

実験のテキストは前任者が残していったものをほぼそのまま引き継いでいますが，「引き継ぎ」作業はまったくなかったので，試行錯誤しながら自分で内容を変えていきました。最初はいろいろな面で教えているこちらのほうがとまどってしまい，学生たちにも迷惑をかけたなと思っています。

しかし今ではかなり慣れてきて，学生の反応を見ながら教え方を変えるなど，柔軟な対処もできるようになってきました。自分はかなり役に立つ実験をしているのではないかという気さえしてきています。

現役でがんばるワンボードマイコン

この授業の仕組みを簡単に説明しましょう。実験は週1回，テーマは全部で12個あり，それをグループごとに順番にこなしていきます。実験は朝10時半に始まり，一応夕方4時半まで。ですが，4時前に終わるグループもあれば5時すぎになるグループもあります。こうして，ひとつのテーマは1日で終了させます。

学生は電気コースと情報コースの2つに分かれていますので，教えるほうには毎週2回，違う学生が10人ちょっとずつ送りこまれてくるわけです。これに加え，昨年10月から12月にかけては夜間部の学生も受け持っていたので，教えた時間数にしたらすごいものでした。

しかも実験時間中に指導すればそれで終わりというわけではなく，レポートの採点および再提出の処理などがあります。レポートを受け取っても，一定の点数以下の学生は個別に呼び出して再提出させるシステムになっているからです。これは採点する側の親心ともいうべきもので，「悪い点をつけるのはかわいそうだから少しでも上げてあげよう」という考えに基づいています。

僕が担当する実験の主旨は，計算機とはいったいどういうものかを，自分の手で触り，自分の目でCPUを見て実感してもらうことにあります。

使用するのはTK-85といういかにも古めかしく見えるワンボードマイコン。当然パソコンとは違い，1枚の基盤がむきだしになっているもので，その上に碁盤目のキーボードと8文字分のLEDがあります。もちろんCRTはついていません。電源以外はそのボード1枚で閉じているわけです。「オールインワン」ですね（うー，懐かしい響き！）。

CPUは8085です。8ビットのCPUといえば，今でこそZ80が最も有名といえましょうが，それがまだ普及していなかったころは8080が一世を風靡していました。8085という石は，8080の強化バージョンとでもいうべきものです。ただし命令数の変化としては，たった2つ増えただけです。

Z80と比較すると，たとえば相対アドレッシングができない，ループ専用命令がない，などいろいろとプログラミングには苦労が伴います。しかし，だからこそ実験には向いているといえるでしょう。

パソコンに対してつい陥ってしまう類の印象，つまり「箱の中身はどうなっているのかわからないが，フロッピーを差し込むと動作するもの」というようなあいまいなイメージを，こうした「むきだし」の機械によって取り払おうとしているわけです。

電卓なら一発なのに……

さて，肝心の実験の内容ですが，大きく次の2つに分かれています。

実験1

8ビット×8ビットの掛け算をするプログラムを作って入力し，それをトレースモードで各レジスタ値の変化を追いながら実行する。次に，サブテキストに載っている掛け算をする2つの参考プログラムを解読し，自分の作ったプログラム，およびこの2つのプログラムの長所，短所を検討する。

実験2

いくつかの命令（比較命令，レジスタペア加算命令，シフト命令など）について個別に実行し，その前後でのレジスタやフラグの変化を調べる。特に各フラグがどういうときにセットされたりリセットされるかを詳細に検討する。

実験1では，自分でプログラムを作成せずに参考プログラムのどちらかを採用しても構わない，としているため，(残念ながら)まず自分で作るグループはありません。

この実験のポイントは（レポート採点の基準になるといっているのですが），なぜ入力したプログラムで掛け算が実現できるのかを理解することであり，当然レポートにおいても自分の言葉でそれを表現してほしいわけです。

実験2では，機械語のプログラミングにおいてはフラグという概念が大切なので，各フラグの意味を手作業で確認して理解することがポイントです。具体的には，たとえば演算結果がちょうどゼロになるようにレジスタを設定してからその命令を実行し，ちゃんとZレジスタが1になっているか，などとひとつずつフラグの機能をチェックしていくことを意味します。ここでは，たとえばキャリフラグは引き算命令ではどんな意味を持つのか，定義されていないが変化するようなフラグに規則性はあるのかな

図1　電気系工学実験のテーマ

1．直流ブリッジによる抵抗の測定
2．直流電位差計による電圧の測定
3．交流ブリッジによるインダクタンス，キャパシタンスの測定
4．半導体素子の静特性
5．整流回路の実験と製作
6．交流回路の電力特性
7．LCR回路の周波数特性と過渡特性
8．等電位分布の測定
9．電磁誘導の測定
10．マイコンの入出力
11．データの表現と演算
12．マイクロコンピュータと機械語

（今となっては10，11，12以外はほとんど内容がよくわからぬ）

どが検討すべきテーマとなってきます。

ところが，学生たちは，普段から計算機に接している（あるいは興味がある）人ばかりではありませんから，なかなかこちらが期待するとおりにはいきません。

プログラム（つまり機械語の羅列）とアルゴリズムのギャップを埋められない，命令に付随するものとしてのフラグの持つ意味の理解に至らない，などの壁をなかなか崩すことのできない人が少なからずいます。

ブラックボックスに光を

さて，実験の1週間後に締切りとなるレポートには，さらに難しい「検討事項」なるものを記さなくてはなりません。これがまた，僕がいうのもなんですが，真面目にやろうとする人にとっては少々難しすぎるものなのです。それは，たとえば次のような文章で出題されます。

「各命令について命令フェッチから実行までのゲート信号のシークエンスについて説明せよ」

つまり，ある命令をメモリからCPUに取ってきて実行を終えるまでに，CPU内でどのような信号が順番に出されるかを，CPU内のデータの通路(パス)にある門(ゲート)の開閉に即して説明せよ，ということです。

一応テキストには，80系のCPUの構成図と名のついた，レジスタやALUの書かれた簡単なブロック図だけは載っています。まず，その図のどこにゲートがあるかを自分で仮定することから始めなければなりません。それからCPU内におけるデータの流れを考えてそれぞれのゲートを開けたり閉めたりするわけです。

この検討課題はちょっと難しいので，こちらでかなり説明する必要があります。手間は取りますが，なんとかわかってくれるようです。僕は，この検討課題にこそこの実験テーマの持つ大きな意味があるのではないかと思っています。実験のテーマは必ずしも機械語のプログラム方法だけではないのですが，機械語プログラミングの教え方としての側面を考えてみましょう。

CPUをブラックボックスとして，命令の意味とレジスタ構成だけを教えるというのもひとつのやり方だと思います。でも，その命令がCPU内で実際にどのように実現されているかを教えることは，単に命令セットを知っている以上の深い理解に裏打ちされたプログラミングへ導くことにつながると思うのです。

一般に，あることをよく知ろうとするには，無駄に見えてもその一段下層のレベルのことをじっくり理解することが重要であるといえるのではないでしょうか。

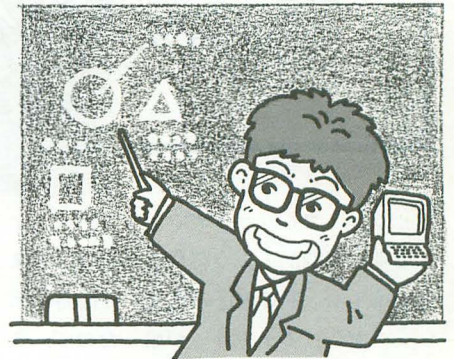
母国語を自由に選択できる権利

実験では，学生たちになんとかCPUの仕組みや原理をわかってもらおうとします。しかもそれは「アルゴリズム→機械語プログラム→各命令→CPU内の処理内容」というように，10年以上前のフォン・ノイマン型の計算機に肉薄していくという方法論に基づいています。

一方，その実験を担当していないときには，そうしたフォン・ノイマン型計算機からの脱出を懸命に試みようとしているのですから，滑稽といえばこれほど滑稽なことはないかもしれません。

大事なのは，何を教えることが将来その学生にとって役立つか，ということでしょう。たとえば，流行のめまぐるしい計算機言語などは，どの言語をまず必修として知ってもらったらいいか，と教える側でもよく話題になります。

自然言語は，単なる情報を伝達するメディアではなく，人の頭の中の思考・思想と予想以上に密着しているものです。同様に計算機言語もその取り扱う得意なテーマは大きく異なってきます。

個人がどの自然言語を母国語として持つかは，生まれた場所によってほぼ決まって

しまいます。しかし計算機言語にはそうした制約はありません。ですから大学でも，必修などで一律に決めるのではなく，できるだけ選択の幅を広げたほうがいいのではないかと僕は思っています。

最初に知った言語はprologやGHCだ，Cなぞ知らん，というような新人類こそ新しいことを言い出してくれることでしょう。計算機アーキテクチャでも，フォン・ノイマン型計算機でしか成り立たない話をどこまで相対化して教えていいのかということは，この先考えなくてはならないことです。

共通1次試験のあった土曜と日曜は，試験監督でたいへんでした。真剣な受験生たちの緊張感がこちらに伝染して肩が凝ってしまいましたので。ちょうど10年前に共通一次試験は誕生し，僕はそのときの試験を受けて大学に入りました。そして，この形で行われる最後の共通一次をまた体験できたわけです。なんとなく感慨深いものがあります。僕の代から今回の試験を受けた学生たちの代まで，この試験はいったいどのような学生を生み続けてきたのでしょうか。

教員生活を垣間見る

学生は本誌を参考書として持ってくる

学生の中には，機械語の一覧表が載っている本を参考書として持ってくる人がいますが，その中にOh!MZを持参した人がいたときはちょっと感動しました。かなり前の機械語入門の特集で，その学生はそれを見ていったんZ80でプログラムを作り，そして8085用に直しているようでした。

学生を（悪いけど）分類する

学生のタイプを2つの尺度で分類すると次の4種類になります。

a）　計算機の知識－多，学習意欲－大
b）　計算機の知識－多，学習意欲－小
c）　計算機の知識－少，学習意欲－大
d）　計算機の知識－少，学習意欲－小

d）のタイプにはもちろん教える側は苦労しますが，予想外に困るのがb）のタイプです。このタイプは，グループ全体のやる気を削いでしまうという傾向があるようなのです。つまり，身近に知識の豊富な奴がいて，そいつが適当にやればなんとかなるさという構えでいると，周りにいる者もやる気がなくなってしまうというわけです。

ついに出現！　驚異の人間複写機

折りにふれ，冗長なレポートを書くな，必要十分なレポートを書け，といっています。その甲斐あってか(?)，ついに出現したのです。驚異のレポートが。テキストおよび配布した参考資料のサブテキスト（20ページ以上）の1字1句，細かい表や図も含めてそのまま写したものが。ご苦労さまと頭を下げたくなりました（おいおい！　ところが時間は有限なのだよ）。

第33回

猫とコンピュータ

またの名をグルメ猫

Takazawa Kyoko
高沢 恭子

冬。ウールマークのホンニャアは寒さには強いのでしょうか。それはさておき、案駄婆さんをいろんなとこで応用するのって楽しそうですね。たとえば? トオル君とキョウコさんはミシンに入れたいそうですよ。

ホンニャアがソファでぐっすり寝入っている間に「昭和」という時代は終わってしまったらしい。FM放送のさなかに臨時のニュースを告げるチャイムが鳴ったので，テレビのスイッチを入れてみたら，官房長官が毛筆で書かれた額入りの2つの文字を，神妙な顔で披露していた。のどかで少し古風な2文字だ。

伝える内容はそれがすべてだったから，私とトオルはしばらく画面をポカンとながめてスイッチを切った。

土曜日でちょうど会社から帰ったばかりのパパは着替えのとちゅうだったけれど，「平成だって，お父さん」と言うトオルに，「フーン，なるほど……」と言っただけで，「少し会社の仕事があるからね」とそのままマシンルームに行ってしまった。

たしかに，あわてなくても明日から元号はずっとこの呼び方になるというだけのことだし，無理に使わなくても済むものなのだ。

すぎ去った歳月に○○時代と名づけるのはわりあいやさしいけれど，これから来る日々に名をつけるのは占いのようにむずかしそうだ。同じ名をつけるにも，赤ちゃんやペット，船や列車やパソコンの新製品なんていうのは当然ながらみんな顔がある。顔のわかりにくい“これからの日々”に，先んじて冠をつけるくらいむずかしいことはなさそうだ。

4ケタと2けた

西暦だけにしてしまえばよいという人もたくさんいる。そのほうが国際的にも通用するし，じっさいビジネスで海外と日本を行き来している人たちには，元号なんて無縁のものだと思う。

そういう私たちだって，知らず知らずのうちに西暦併用が身についているし，新聞雑誌などの刊行物にも西暦表記は多いみたいだ。よく言われる「日本の国際的な地位の向上」からも，それは自然な移り変わりなのだろう。

「昭和」も60を数えるころになると，「60年代のポップス再流行」なんて聞いたときに，省略して下2ケタで呼ぶ西暦との区別がめんどうな気がしたこともあった。いっそひとつに絞ってくれたほうがいいのかなとも思う。

でも，この日本特有の元号，「もうひとつの呼び名」も，うまく温存しておけるなら，それなりの役割があるだろう。国内のある時代を指して呼ぶときとてもわかりやすいだろうし，西暦とは別に日本独自の歩みをカウントできる分類があるのも，その起因になることの問題は別として，特色といえるのではないだろうか。

それに，私のように知らないウチに西暦の4ケタの数字がいくつか入れ替わってしまううっかり者には，もうひとつの数え方は数字も小さいし，ありがたい救済手段になってくれる。

改元三段跳び

昭和生まれのホンニャアが，翌日，新しい元号になって最初にやったのは，キャットフードにスキヤキフリカケをかけて食べることだった。

最近，いつものラベルのネコ缶に良い顔をしなくなっていたのだが，買い置きがまだ1ダース以上もあって私は憂鬱だった。そこでその場しのぎにフリカケをバラまいてみたら，香りが抜群だったせいかとびつくようにしてむさぼり，お皿をなめるまで食べつくした。

その日の2度めのゴハンでも，ホンニャアは当然のようにフリカケを待った。以前のままの缶詰めフードには後ずさりして横を向いてみせた。おろかな私は，これからずっとフリカケ乗せのキャットフードをホンニャアに食べさせていくのだろうか。

元号が変わる時というのを体験したのはもちろん家じゅう初めてのことだけれど，S市にいる夫の父，ロクロウヘイおじいちゃんは「大正」，「昭和」と目撃し，こんどで3度めということになる。昭和に改元された年には，S市がS町だった時代の町会議員に最年少で初当選をしたそうだ。

そのおじいちゃんが「平成」に改められて初めてやったのは，いつもの散歩のとちゅうでタクシーにはねられることだった。

ゲタばきで悠々と，しかも姿勢のよい早足で歩く背の高いおじいちゃんだが，自宅近くの川沿いの道でうしろからドンとはねられ，ボンネットの上に乗ってしまった。次にそこからヨイショと降りて，車から出てきて蒼白になっている運転手さんに，「Don't mind，だいじょうぶだよ」と手をあげ，さっそうと帰宅した。このことはしばらくしておじいちゃん自身が話すまで誰も知らなかった。

若き日はやがて町長もつとめ，いくつもの企業を誘致するなど，経済，文化発展のリーダーのひとりとなって，町から市へと地域を育てあげるための非凡な力を発揮した。自身も東京に会社を持つなど長い活躍を続け，ごく近年ようやくS市に落ちつく生活になったところだ。

ただしおじいちゃんの勤勉の習慣は崩れない。もともと博学であらゆるカンにすぐれているので，興味を持ったことがらは深く研究してやまないのだ。だからいままでも現在も，計画と抱負が途絶えたためしがない。そしていつでも，おおやけの人々の利益になることに頭をひねっている。おじいちゃんはいまでも，毎日何人ものひとたちに会うのが日課だが，自らの健康への不安や死については語ることをしないので，

みんながおじいちゃんに会うと爽快な気持ちになる。

2年くらい前に増築した20畳ほどのおじいちゃんの書斎には，ワープロもコピー機もある。今年も中国人の先生お墨付きの自作の漢詩に，これまたすばらしい邦訳の詩を添えた年賀状が送られてきた。

無用なことにこだわらず，人の話はよく聞いてやり，必ず良い点を探してほめてくれる。まわりの者はこのはげましで，無い力までわいてくる。おじいちゃんは月に1度くらいゴルフに出かけるが，いつかは若い人でもその時期は避けるという真夏の炎天下でプレイして，ちょっとぐあいが悪くなり入院したこともあった。が，いまは快調だ。

新しい時代をまたひとつ迎えて，おじいちゃんの，時代を見る新たな姿勢と息吹を少しでも多く学びたい気持ちである。

内蔵ばあさん

X68000ACE HDで内蔵ハードディスクから起動するように設定したら，ゲームのフロッピーディスクが立ち上がらなくなってしまった一件は，同じトラブルに関する質問が昨年のマイコン誌6月号の「なんでもQ&A」のページに掲載されていた。よくある質問のひとつらしい。もちろん，ハードディスクタイプが出されたときにOh!Xでもきちんと解説されている。

ハードディスクの設定をする前は，フロッピーディスクドライブから優先的にシステム起動するようになっていたものが，システムをハードディスクへ転送したあとはプライオリティがハードディスクからになるためだそうだ。

あの時はびっくりして，ハードディスクに転送した内容を元に戻したりしたが，解決の方法もいくつかわかったので，もう一度転送のしなおしをした。そして，いつもフロッピーディスクから起動したいならSWITCHのコマンドでBOOTのパラメータをSTDに書き換えればよいというので，そのとおりにしてみた。

めでたくゲームもスムースにはじめられるようになったし，Z'sSTAFFよりひとあし先にSOUND PRO-68KとMUSIC PRO-68Kが手に入ったので，トオルと親友たちは大喜びだ。部活のない日はまずマシンルームでひと遊び。MUSIC PRO-68Kは音楽の法則なんか無視して，並べた音符をメチャクチャに演奏してくれるので面白くてたまらないらしく，例によって笑いころげてすごしている。

ゲームももちろん大さわぎの楽しみようだ。サンダーフォースIIも，待ちかねた源平討魔伝も，吸い込まれそうにきれいな画面なのだ。トオルは5年生のころに熱中したゲームセンターの源平討魔伝がそのままわが家でやれることに感激しきりである。とくにあの案内役の「案駄婆（あんだばぁ）」さんが時々かける，励ましともさげすみともつかない声がゲーセンをほうふつとさせる。「諸行無常よのう」，「先は長い」，「たわむれは終わりじゃ！」，命の灯を示すロウソクが数少なくなると「風前の灯！」。成績不振でゲームオーバーになると「お前の力はそんなものか」。

弁慶も自分が負けると「これで勝ったと思うなよ」なんて言う。これをコインを入れずに存分に味わえるのだ。でもそのあたりが，いちばんだいじなスリルをうばっているかもしれない。

グルメパソコン?

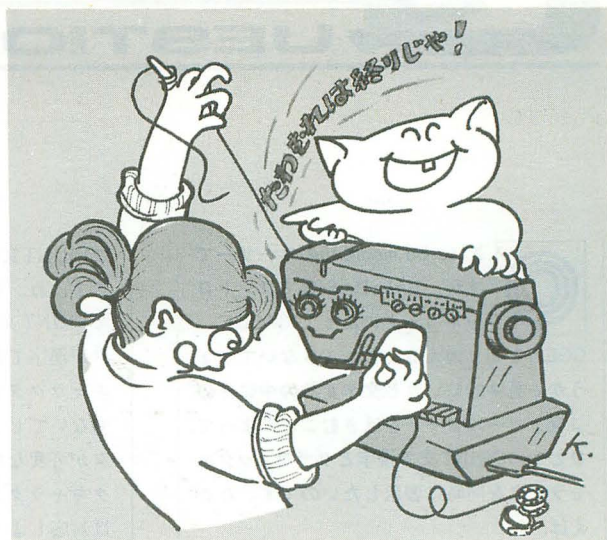
昭和の最後の10年はコンピュータの時代だった。とくにわが家はトオルの誕生と同時にパソコンライフが発展していき，とうとう私がマシンを購入するところまできた。家の中の電化生活を全体的視野で見ると，あまりにパソコン分野の超リッチ（?）なのに比べて，ミシンなど必需品に不均衡が目立つようで少し気になるほどだ。

「こんどはミシンを新調しようかな」とつぶやくと，夫が，

「必要もあまりないのに，複雑すぎる機能は故障の原因になるだけだから，シンプルなものがいいよ」

とこれはまたパソコンマニアとは思えない発言をした。その上，

「ママは直線縫いでじゅうぶんでしょ」

そりゃ，おもな作品は直線と平面を組み合わせた「新学期のぞうきん」くらいだけどあんまりじゃないかしら。

「いまはみんなマイコン内蔵のミシンばっかりよ！　直線縫いだけなんて探したってありませんからね」

するとそばから，

「どうせなら案駄婆が入ってるミシンがいいね，お母さん」トオルが言った。

「“これで縫えたと思うなよ”とか，下糸がなくなりそうになったら，“風前の灯”とかサ。糸がからまったら“たわむれはこれまでじゃ”なんていって，調子にのっていると“今にみておれ”，失敗したら“おろか者め！”なんてネ。楽しくってきっとお裁縫やめられなくなっちゃうよ！」

これは冗談ぬきにして面白そうだ。あまりマジメな機械が多すぎるもの。

X68000もなんとなくその持ち味がわかりかけてきた。Z'sSTAFF PRO-68Kはまもなく手に入る予定だ。その前にすこし本体付属の日本語ワードプロセッサを試してみよう。“先は長いぞ！”。

開拓者たちは未熟なマシンで苦労を重ねしかもたくさんの投資をして今日に至り，後から来る者のほうは，たとえ知識はおぼつかなくとも，やすやすと最新の機種にありつける。まあ，何の道もこんなものかな。

ともかく迫力満点の最新マシンを迎えて昭和を終えたわが家の風景だ。ひとつの悩みは，いまだにマウスというシロモノの扱いにモタついて，手間どっていることだ。

立派なキーボードがありながら，その上マウスなんて，ちょっとネコ缶にフリカケのホンニャアのゴハンを連想したのはまちがいだろうな，ヤッパリ。

QUESTION and ANSWER

X1turbo model 10のユーザーですが，turboBASICでコントロールコードによってKMODE, CGENを切り換えることはできないでしょうか。具体的にいうと文字変数の中にそのコントロールコードをはさむことによって，ひとつのPRINT文で漢字とグラフィックキャラクタを同時に表示したいのです。たとえば，

10　A＄＝“●○”＋CHR＄(××)
　　　　　　　　＋“漢字”

20　PRINT　A＄

とすることによって，

●○漢字

と表示されるようにしたいのですが。

BASIC のコントロールコードにはダミーがいくらかあるようですが，これを利用してなんとかできないでしょうか？

岩手県　藤原　裕也

はっきりいっちゃうとKMODEやCGENを切り換えるコントロールコードなんてものは存在しません。したがって無理です。――なんて答えちゃうとあとが続かないのでここはひとつ，いいアイデアをお教えしましょう。

上にも述べたとおり，まともな方法ではひとつの PRINT 文でグラフィックキャラクタと漢字の両方を表示するのは困難です。もちろん PRINT 文をひとつに限らなければ“●○漢字”のように表示することは容易です。なぜならturbo BASICのKMODE命令は全画面に対する漢字/非漢字表示命令ではなく，これからPRINT されるキャラクタに対する漢字/非漢字表示命令だからです。

たとえば KMODE 1で画面に“漢字”と表示したとします。ここでKMODEをKMODE0に変えても表示された“漢字”という文字はグラフィックキャラクタに変わったりはしません。これを利用してやれば次のようにして“●○漢字”と表示することも可能です(初期状態を KMODE0とする)。

10　PRINT　“●○”;
20　KMODE　1
30　PRINT　“漢字”
40　KMODE　0

しかしこれではPRINT文をひとつにすることができません。実際使う分には絶対的にPRINT文がひとつですむほうがラクなんですよね。なぜかって？　前記のようにしてPRINT文を処理すると，あらかじめデータが漢字であるか，あるいはグラフィックキャラクタであるかを知っていなければならないでしょう。すなわち，表示するデータが可変な場合（漢字であるかグラフィックキャラクタであるかわからない場合）には対応しようがないのです。

さて，そこでどうするかです。もちろんまともな方法では不可能です。ここではあなたが考えてみたようにダミーのコントロールコードを使ってみることにします。

突然，話は変わりますがRS-232C通信における漢字の取り扱い方について知っていますか。RS-232Cでは漢字の送受信に2つのやり方があります。すなわち，

J：漢字インコードKI(＆H1B4B)で日本語文字列の始まりを示し，漢字アウトコードKO(＆H1B48) で日本語文字列の終わりを示します。

N：シフトJIS コードを使用します。

という2つのモードがあります（マニュアル参照のこと)。

ちょっとわかりにくいかもしれないので少し説明しましょう。まずJモードです。このモードでは初めにくるデータはすべて非漢字として処理します。しかしある特定のデータ(KI)がくると以降のデータを漢字として処理します。そしてKOコードによって再び非漢字モードに戻ります。Nモードのほうは単純で，データをすべて漢字モードとして扱う（すなわちグラフィックキャラクタは送受信できない）のです。

話をもとに戻しましょう。RS-232Cと漢字表示のどこに共通点があるんだとお思いの方もいらっしゃるかもしれませんが，大ありなのです。

RS-232C での2つのモードを漢字表示に置き換えてみましょう。いうまでもなくNモードはKMODE 1の固定モードですね。それじゃあJモードは？　このモードはKMODEが可変です。すなわちあるコードによってPRINT 文自体が処理を変えているのです。

鋭い人はもうおわかりでしょう。そう，PRINT文の代わりに，あるコードによって処理を変えるPRINT ルーチンを使ってやればよいのです。そして「あるコード」にはダミーのコントロールコード(BASICで使用されていないコントロールコード）を使ってやるというわけです。

詳しく説明しましょう。結論からいうと，PRINT文の代わりにPRINT 文の代わりをするルーチンを使うのです。ここではそのルーチンを“プリント”としましょう。

たとえば漢字とグラフィックキャラクタの混在する文字列A＄があったとします。ただしA＄にはそれなりの処理を施しておく必要があります。すなわち漢字部分の前と後ろに特定のコードを埋め込んでおくのです。ここでは漢字部分が始まることを示すコード（漢字インコード）をKI，漢字部分が終わることを示すコード（漢字アウトコード）をKOとしておきましょう。

すなわち，“●○漢字”と表示するためにはA＄は，

A＄＝“●○”＋CHR＄(KI)

リスト1

```
10 'Kanji/Graphic chr.
20 '         for X1turbo
30 '
40 KMODE 0: KM=0
50 K$(1)="▌ソ■r": K$(2)="▃█▃K▃V▃B": KC$=CHR$(&H18)
60 A$="●○"+KC$+K$(1)+KC$
70 P$=A$: GOSUB "ﾌﾟﾘﾝﾄ"
80 B$="▁▃█"+KC$+K$(2)+KC$+"█▃▁"
90 P$=B$: GOSUB "ﾌﾟﾘﾝﾄ"
100 END
110 LABEL "ﾌﾟﾘﾝﾄ"
120 FOR I=1 TO LEN(P$)
130 PP$=MID$(P$,I,1)
140 IF PP$=KC$ THEN KM=1 XOR KM: KMODE KM: GOTO 160
150 PRINT PP$;
160 NEXT: PRINT: RETURN
```

＋"漢字"＋CHR＄(KO)
となるわけです。

さて，このA＄を"プリント"ルーチンを使って表示してやるわけです。ルーチンに汎用性をもたせるためにルーチン内で使う表示文字列変数をP＄とし，コールする前にP＄に表示したい文字列A＄を代入しておきます。

すなわち，

```
10 P＄＝A＄
20 GOSUB "プリント"
```

のようにしてコールするわけです。

問題は"プリント"ルーチン内でどう処理をするかです。実はこれは非常に単純です。まず，表示文字列P＄のうち左から1バイトずつ取り出して表示していきます。初期状態はKMODE 0ですからグラフィックキャラクタももちろん表示できます。

問題は漢字表示のときですが，漢字部分の前には必ず漢字インコードKIがありますから1バイトずつ取り出していく過程で漢字部分が始まることを検知することができます。ですから，KIコードを検知したらすばやくKMODE 1に切り換えてやればよいのです。同様にしてKOコード(漢字アウトコード)を検知したらKMODE 0に戻してやればいいというわけです。

実際にプログラムにしてみたのがリスト1です。リスト1ではKI/KOにダミーのコントロールコードである18Hを使っています。このようにKIとKOを同じコードにしてもまず問題はないはずです。むしろ処理はラクになるしね。

ちなみにリスト1は漢字/グラフィックキャラクタ混在のために変なリストになっていますが，K＄(1)/(2)はそれぞれ"漢字"/"レガシィ"という全角文字です。もっとも今までやってきたのと同様な考え方をすればリストでも漢字/グラフィックキャラクタの混在が可能なんですけどね。でもこれは自由課題。

このように一見不可能なことでも発想次第でなんとでもなるものです。BASICはあくまでツールであり，利用するものにすぎません。だからBASICでは無理などといわずに，BASICを精一杯利用してなんでもやりとげてみましょうよ。

S-OS"SWORD"について質問します。私はMZ-2500(2DD)を使っていて，MZ-2000モードでも160トラックまでフォーマットできるはずだと思い，＃MXTRK(1F66H)を160(A0H)に書き換えてディスクをフォーマットしたのですがうまくいきません。

確かに，Dコマンドでディレクトリを見ると4AH Clusters Freeとあるところが7AH Clusters Freeに変化しているのですが，Sコマンドでセーブしていくと30H Clusters Freeのところでそれ以上セーブせず，Bad Recordと表示されます。

＃MXTRK以外にもどこか書き換えないと160トラックまで使用できないのでしょうか。　静岡県　池田　敬士

結論からいうと＃MXTRKを書き換えただけではうまくいきません。これには理由があります。その原因はMZ-2000用のS-OSのディスクI/Oルーチンにあります。このソースリストを持っている人は見てみるとわかりやすいと思いますが，ディスクI/Oルーチン中のRADJというサブルーチン(2B2FH～)に鍵は隠されています。

すなわち，このルーチン中でレコードナンバーの最大値が1279(4FFH)に制限されているのです。1279というのはディスクが2Dの場合の最大値で，当然2DDや2HDの場合はこれを大きく超えてしまいます。

ですから，このディスクI/Oルーチンを利用して2DDにアクセスするためにはまずこのサブルーチンを書き換えることが必要となるわけです。具体的には2B3BHからの2バイトにレコードナンバーの最大値を書き込めばよいのです。2DDの場合は16×160－1＝2559となります。

MZ-2500のS-OSでは2B97H以降にあるディスク割り付けテーブルに従ってちゃんと最大トラック数/最大レコード数などを自動的に書き換え，さらに自分にパッチを当てて2D/2DD/1DDの違いに応じアクセスの仕方を微妙に変化させています。どうしてもMZ-2000用のディスクI/Oルーチンで2DDを使うんだという人はMZ-2500用を参照し自分でトライしてみてください。

もうひとつの理由は"SWORD"の心臓部ともいえるDOSモジュールの中にあります。S-OSがカセットテープやQD，ディスクのタイプを気にせずにまったく同じルーチンで入出力を行えるのは，DOSモジュールのおかげなのですが，このDOSモジュールでは設計上，7Fクラスタまでのドライブしかサポートできないのです。

ですから，MZ-2500では，640Kバイトの容量を持つ2DDのディスクも512Kバイトまでしか使用できません。

ところでハード的に2DDを扱えるS-OSマシンといえばほかにX1turboがありますが，X1turboではどのような処理を行っているのでしょうか。

実をいうと，X1turboではディスクI/OをほぼBIOS ROMに頼っているので，MZ-2500のようにあまり考える必要がなくてすんでいるのです。これはS-OSとHu BASICのディスクフォーマットがほとんど同じだからできることでもあります。

とはいえども，BIOS ROM内のディスクI/Oルーチンはちゃんとディスクの割り付けデータに従ってアクセスの仕方を変えているのですからやっていることはMZ-2500とほぼ同じです。ただ，X1turboのほうはそれが表に出てこない，というわけです。

(華門　真人)

質問にお答えします

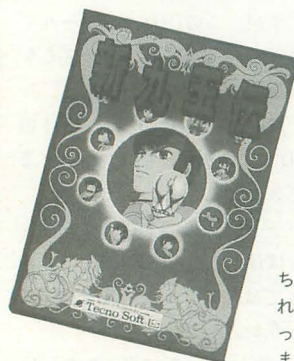
日ごろ疑問に思っていること，どんなことでも結構です。どんどんお便りください。難問，奇問，編集室が総力を上げてお答えいたします。ただし，お寄せいただいているものの中には，マニュアルを読めばすぐに回答が得られるようなものも多々あります。最低限，マニュアルは熟読しておきましょう。質問はなるべく具体的に機種名，システム構成，必要なら図も入れてこと細かに書いてください。また，返信用切手同封の質問をよく受けますが，原則として，質問には本誌上でお答えすることになっていますのでご了承ください。なお，質問の内容について，直接問い合わせることもありますので，電話番号も明記してくださいね。

宛先：〒102 東京都千代田区
九段南2-3-26井関ビル
㈱日本ソフトバンク出版部
「Oh！X質問箱」係

愛読者プレゼント

プレゼントの応募方法

とじ込みのアンケートはがきの該当項目をすべてご記入のうえ，希望するプレゼント番号をはがき右下のスペースにひとつ記入してお申し込みください。締め切りは1989年３月18日の到着分までとします。当選者の発表は1989年５月号で行います。

テクノソフト　☎0956(33)5555

新九玉伝

X1turbo用
5"2D版　8,800円

2名

ちんねん，そんねんの手により九玉が封印されて数百年。平和な日々を送っていた人々だったが，ある日何者かが再び封印を破ってしまった……ご存じ，コミカルRPGの続編。

2

コスモス・コンピューター　☎03(770)1821

ウォーニング

X68000用 5"2HD版
7,800円

2名

７つの惑星を駆け巡るＳＦウォーシミュレーションRPG。貿易交渉あり，宇宙のならず者との戦闘あり，謎のアイテムや禁断の兵器ありで盛りだくさん。

アクセス　☎03(233)0200

X1エミュレータ

X68000用 5"2HD版
9,800円

2名

X68000上でX1シリーズのソフトを利用できるソフトウェアエミュレータを２名に。X1の２DファイルをX68000用にコンバートするユーティリティ付き。なお，このソフトの利用にはX1の本体が必要です。

4

サン・ミュージカル・サービス　☎03(419)8839

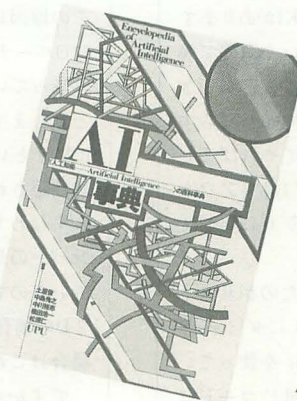
Musicstudio PRO-68K対応

ソングファイル68Kシリーズ

a) 国本佳宏ソングファイル
b) 佐久間正英ソングファイル

5"2HD版　5,800円
各１名

Musicstudio PRO-68K用のオリジナルデータ曲集。ローランドのMT-32の音色に対応。

ユーピーユー　☎03(222)1501

AI事典

7,800円

2名

思想・哲学から言語学・論理学・情報工学に至る広範な知識を１冊に詰め込んだ人工知能の百科事典。読み物としても楽しめる。

1月号プレゼント当選者

1Master of Monsters（青森県）今井慎一（神奈川県）杉崎伸一（茨城県）倉田泰幸（愛知県）諸鍛治稔勝（愛媛県）豊崎剛　2第４のユニット（東京都）小俣裕司（北海道）遠藤直樹（福岡県）田中了嗣　3TETRIS（東京都）渡辺一雄（静岡県）小林郁夫（高知県）東谷隆英　4『X68000パワーアッププログラミング』（東京都）青木孝文（茨城県）隈田原太（京都府）吉井照昌（兵庫県）橋本浩二（富山県）木村浩之

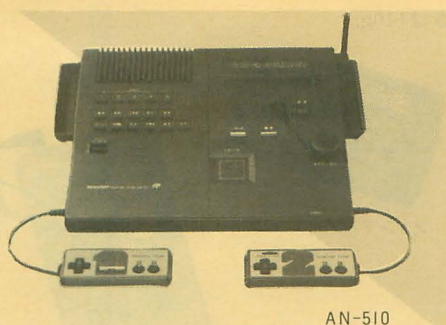
（敬称略）

以上の方々が当選されました。おめでとうございます。品物は順次発送いたしますが，入荷状況などにより遅れることがあります。また，公正取引委員会の告示により，このプレゼントに当選された方は，この号の他の懸賞には当選できない場合がありますのでご了承ください。

PENGUIN INFORMATION CORNER

ペ・ン・ギ・ン・情・報・コ・ー・ナ・ー

NEW PRODUCTS

ファミコンタイトラー
AN-510
シャープ

AN-510

ファミコン用ソフトが使用できるゲーム機能に加え，キャラクターつきのムービータイトル画面を内蔵したファミコンタイトラーAN-510がシャープから昨年末に発売になった。価格は43,000円，ビデオ，音声，S映像入出力端子つき。

内蔵のタイトル画面は，アニメーションつきのものを含め全部で75種類。子供の成長記録や家庭行事の記録用などを中心に揃えてある。タイトル画面を呼び出して映像に重ね合わせるだけの簡単操作。

また，オリジナルのタイトル作成用に漢字約1000文字，ひらがな，カタカナ，英数字，および書体も太字やマンガ字など4種類を持っている。文字は付属のペンで本体入力部に書くだけ(手書き文字認識)。

さらにメモリ機能で5タイトルまでワンタッチで挿入でき，コントローラ2に内蔵マイクでナレーションを入れることも可能。

〈問い合わせ先〉

シャープ㈱　☎06(621)1221,03(260)1161

ワイヤレスヘッドホンステレオ
「ビーイング」JC-T70
シャープ

シャープは，ワイヤレスヘッドホンステレオJC-T70を2月10日から発売した。価格は32,800円。

JC-T70の特徴は，本体にWループアンテナを採用し，送信指向特性を向上させたことと，送受信周波数の2局切り換えで他のワイヤレス機器との混信を防ぐようにしたこと。

内蔵充電池と着脱乾電池を併用した場合，約9.5時間の再生ができる。

〈問い合わせ先〉

シャープ㈱　☎06(621)1221,03(260)1161

JC-T70

低価格インテリジェントモデム
GSM2400
関東電子

関東電子は，エンドユーザー向けの韓国金星社製のインテリジェントモデムGSM2400（29,800円）を発売開始した。同社は今後もNIES商品の取り扱いを進めていくという。

GSM2400は全2重2400bps，ヘイズATコマンド，CCITT/BELL規格対応。

電話番号は5カ所まで，1件につき最大33文字登録できる。

〈問い合わせ先〉

関東電子㈱　☎03(257)6291

GSM2400

インテリジェントモデム3機種
マルチモデム224EHI/696EH/V32
コア

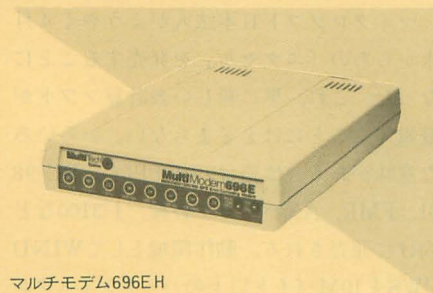
米Multi-Tech Systems社製のインテリジェントモデムMulti Modemシリーズ3機種が発売開始された。3つとも通信プロトコルにMNPを採用。

まず，マルチモデム224EHIは全2重2400bps，ヘイズATコマンド，MNPクラス5およびCCITT/BELL規格対応。価格は98,000円。

696EHは9600bps，MNPクラス6に対応し，最高実効速度は19200bpsを実現。価格は185,000円。

最後に，同じく9600bps，最高実効速度19200bpsのV32は，MNPクラス5に対応，全2重で一般電話回線のほか2線式専用回線で通信が行える。価格は360,000円。

〈問い合わせ先〉

㈱コア　☎045(411)8861

マルチモデム696EH

サウンドプロセッサつきジョイカード
Joy Card mkII SANSUI Version
山水電気

山水電気は，ファミコン用ジョイカードの新製品Joy Card mkII SANSUI Versionを昨年12月末から発売開始した。価格は2,900円。

この新製品は付属のヘッドホンを通してテレビゲームの音声をステレオで楽しめるのが特徴。もちろんオーディオアンプにも接続できる。また「ジョイスティック連動定位移動回路」によりキャラクターの動き

に合わせて音像の定位を左右に動かすことも可能。

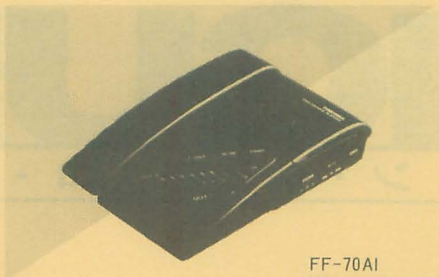
連射機能は2段階切り換えて1プッシュ最大15ショット/秒。

<問い合わせ先>

福島サンスイ(株) ☎0248(75)4134

Joy Card mkII SANSUI Version

音声でダイヤルできる 家庭用電話機FF-70AI 東芝

東芝は，あらかじめ登録した単語を受話器に向かって言うだけで，ダイヤルボタンを押さずに電話がかけられる「キッスホンオフFF-70AI」を3月1日から発売する。価格は30,000円。

FF-70AI

新製品は，話し手の音声を音の高低に合わせて6つの周波数帯域のデジタル信号に変換し，それを，登録してある音声データと比較して認識するもの。

相手先の名前などを19件まで登録でき，番号をよく覚えていない場合や，暗くてダイヤルボタンがよくみえないときなどに便利。

登録した単語を受話器に言うと，最も似ていると認識された単語の登録音声が受話器から再生されるので，相手先が正しいかどうかの判断もできる。

このほか，暗証番号による特定の相手とのメッセージのやりとりも可能。

<問い合わせ先>

(株)東芝 ☎03(457)2100

カード型新製品2種 フォンマスター/VOICE STATION セイコー電子工業

カード型カナ表示電子メモ・フォンマスターと，IC音声英会話カードVOICE STATIONがセイコー電子工業から発売された。

フォンマスターDF-630(6,500円)は，406件の名前と番号を記憶でき，電話番号や口座番号，時刻表など，いろいろな数字情報をインプットしておける。シークレット機能，メモリバックアップ機能つき。重さは110g。

フォンマスター

VOICE STATION

Again Watch

気になるソフト，エクセル登場

マイクロソフト日本法人がようやく4月末からあの「エクセル」を発売することになった。これ，単に新しい表計算ソフトが登場することにはとどまらない。いろいろな意味がある。価格は9万8千円で，PC-9801，FMR，AXパソコン各種，J-3100など向けに発売される。動作環境としてWINDOWSと10Mバイト以上のハードディスクが必要となる。

エクセルについてご存じない方もいらっしゃるだろうから，説明しておこう。アップルのMacintosh用で大ヒット，それを受けてIBM-PC/AT用にMS-WINDOWSのもとで動く表計算として米マイクロソフトが発売してきたソフトである。Mac用が100万本以上,AT用が30万本以上を販売。ATではロータスの「1-2-3」の足元にも及んでいないものの，Mac用の代表的なアプリケーションソフトとしてすっかり定着している。

基本はデータベース機能を強化した表計算ソフトにグラフ作成機能を追加した製品である。こういってしまうと「1-2-3」やMSチャートつきのマルチプランとそう変わらないような印象を受けてしまうが，応用機能であるマクロがかなり強力だ。簡易言語として帳票処理や自動演算処理に向くプログラミングツールとして使用できるのはもちろんのこと，窓を自動操縦したり，他のソフトとのデータ入出力機能を豊富に用意してあることから，かなり複雑なアプリケーションが組める点が特徴といえるだろう。

この「エクセル」にはマイクロソフト日本法人も力を入れており，初年度最低でも5万本は売りたいとの販売目標を掲げている。だがどうだろうか。力があり，名前も通っているとはいえ，日本で売れるかといえば，ちょっと難しい局面にある。というのはMac用とPC/AT用の販売本数の開きを見ればピンときていただけると思う。つまり，Macにははじめからウィンドウシステムがついているのに対し，PC/ATではMS-WINDOWSがあまり普及していなかったからにほかならない。

では日本はというと，PC/ATに状況は近い。PC-9801にせよ，IBM5550や富士通のFMRにせよ，どのユーザーもMS-WINDOWSなんか使っていない。ある人に聞いたところ，日本のWINDOWSユーザーはすでに3万人規模になっているという。これが事実としても，エクセルのターゲットユーザーと完全に一致するわけではないし，ましてや全員が購入するわけではない。

WINDOWSかOS/2か?

したがって今年，マイクロソフトがエクセルを量販しようと思ったら，WINDOWSをも同時に拡販する必要がある。この場合，期待できるのはAXパソコンである。AXパソコンにはMS-DOSver3.2とあわせてMS-WINDOWS2.0をハードにバンドリングしているものもある。これが売れると自動的にWINDOWSも普及する。そして好都合なことに，大部分のAXパソコンはビジネス用パソコンとして販売される。しかもWINDOWS用アプリケーションはほとん

VOICE STATIONはIC音声を使った英会話練習機で，海外旅行やビジネスなどの用途に合わせて3種類が用意されている。価格はそれぞれ9,800円。

収録文例はジャパンタイムスの「英会話手帳」からの抜粋。文例の検索やリプレイが素早く行えて便利。

約1分間操作がなければオートパワーオフする。重さは52g。

〈問い合わせ先〉

セイコー電子工業㈱　☎03(638)5229

INFORMATION

アマチュアCGAコンテスト発表会
プロジェクトチームDŌGA

一般公募による「第1回アマチュアCGAコンテスト」(プロジェクトチームDŌGA主催，アスキー，シャープ協賛）の発表会が2月26日(日)，市ヶ谷のエルムホールにて行われる。同コンテストには，大学のコンピュータサークルをはじめ各地から，X68000などによって制作された数々のCGA(コンピュータグラフィックアニメーション）作品が寄せられており，当日はこれらの作品の上映も予定されている。

日時　2月26日（日）午後2時

場所　シャープ東京支社8階エルムホール（JR市ヶ谷駅より徒歩1分)

〈問い合わせ先〉

プロジェクトチームDŌGA　☎06(321)8735

BOOKS

小説ウィザードリィ 隣り合わせの灰と青春
JICC出版局

「ファミコン必勝本」1988年vol.5からvol.22に連載された作品が，単行本として発行された。ロールプレイングゲーム，ウィザードリィのノベライゼーションである。

小説ウィザードリィ

ベニー松山　著

A5判，272ページ，980円

〈問い合わせ先〉

JICC出版局　☎03(234)4621

三国志ハンドブック
光栄

三国志ハンドブック

シミュレーションゲーム三国志の攻略本。国づくりの基本・権謀術数・人材登用術・兵法大全と，目次を見ると歴史小説みたいなところがいい。イラストや写真もふんだんで読むだけでも楽しめそう。

シブサワ・コウ　編

A5判，208ページ，1800円

〈問い合わせ先〉

㈱光栄　☎044(61)6861

どないという状況をあわせて考えると，エクセルは対象ユーザーといい，タイミングといい，実にピッタリだ。

注）と書いたものの，実際にはAXパソコンの商品化の過程を追っていくと，エクセルの販売を前提にして商品設計された節がある。

ではPC-9801とかFMR, IBM5550とかはどうか。こちらはかなり微妙だ。というのも今年中旬から秋にかけて，プレゼンテーションマネジャーつきOS/2が登場，OS/2も待機中の時代から本格販売の時期に突入する。そうなるとWINDOWSと正面からバッティングすることになり，どうももろもろの声を集めてみると，WINDOWSの分が悪い。エクセルに続くWINDOWS対応アプリケーションもそう相次ぎ発売される様子もない。かたやOS/2のアプリケーションは，続々と(移植も含めて)開発中のようだ。

となると，PC-9801ユーザーがエクセルだけのためにWINDOWSを購入するかということになるのだが，それは極めて少数に過ぎないであろう。結局のところ，エクセルの問題はOS/2対WINDOWSの普及レースにまで話が及んでしまうのだ。

ちなみにエクセルだが，MS-WINDOWSだけなどといわずX68000用も出せばどうだろう。2～3千本の販売は堅いはずだが。

今年は失速？　小型機市場

ちょっと話が変わるのだが，今年のパソコンやワークステーションからなる小型コンピュータ市場はかなり失速するという見通しが欧米では濃厚になっている。

昨年春頃からIBMの新型パソコン，PS/2が失速していることは伝えられている通り。これにはいろいろな理由がある。互換機メーカーが新型の32ビット高速バス「EISA」を共同開発してATを担ぐ計画を打ち出したこともあるし，OS/2アプリケーションの荷動きが予想よりもペースが悪いこともある。ただ結局のところ，道具としてのパソコンを考えた場合，完成度や用途の広がりから総合的にみて，PC/ATを捨ててまでPS/2を買う必要はない，と大部分の人が判断した結果であろう，と見られる。

で，そのATだが，ここにきて需要が一巡した雰囲気が強く，全体的に頭打ちの様相を呈している。

ATやPS/2の心臓部であるi80286やi80386などのマイクロプロセッサを製造販売しているメーカー，インテルによると，需要の成長のピークは昨年夏であり，それ以降の成長率は下降現象が見られるという。たとえば日本法人であるインテルジャパンの売上高も昨年は前年比で78%も成長したが，今年は10%いくかいかないかだという。この傾向は米国本社も同様であり，いずれもPCマーケットの失速を反映している。

またワークステーションもまだそれほど大きな市場になっていないことに加えて，一連のUNIX二派対立が障害になり，それほど大きな成長は期待できないとされている。

マシンの需要が期待できないとなると，マイクロプロセッサだけにとどまらずメモリチップや周辺機器も成長は鈍化する。業界関係者は水面下ではらはらしているのが現状だ。おそらく何も心配していないのは日本電気だけではないだろうか？　(K.T.)

FILES Oh!X

このインデックスは，タイトル，注記——筆者名，誌名，月号，ページで構成されています。寒かったり暖かかったりとヘンな気候が続きます。本誌もそろそろ3月号，気分は春。そろそろ新製品の話題が気になりますね。

参考文献

I/O　工学社
ASCII　アスキー
テクノポリス　徳間書店
POPCON　小学館
マイコン　電波新聞社
マイコン BASIC Magazine　電波新聞社
LOGIN　アスキー

一般

▼ハイパーメディアCD-I
CD-Iについての最新情報を掲載。——川上宏，POPCOM，2月号，226-230pp.

▼BASIC講座
ゲームを作りながらBASICの基礎を教える。——編集部，マイコンBASIC Magazine，2月号，52-55pp.

▼FM音源おもしろセミナー第3回
FM音源で音作りをするうえでの基礎について。——川野俊充，マイコンBASIC Magazine，2月号，60-62pp.

▼コンピュータ・ミュージック講座 その2
シャープのMIDIインタフェイスやMusicstudio PRO-68Kなどを紹介。——山本数生，マイコンBASIC Magazine，2月号，63-65pp.

▼特集　国際科学技術娯楽年間
CD-Rやデータグローブなどの最新技術とパソコンの関係をホビー的視点から予測。——編集部，LOGIN，1・2号，226-239pp.

▼THE NEWS FILE!
CD-Iソフトの紹介や，世界OS戦争の話題など。——編集部，LOGIN，1・2号，266-273pp.

▼パソコン通信
音楽とパソツーその2と題してロック情報のネットを紹介している。——編集部，LOGIN，1・2号，278-281pp.

MZ-80K/C/1200/700/1500

MZ-700/1500

▼蚊ッ蚊ッ蚊ッ！
1分間にどれだけの蚊を叩けるか競うゲーム。——神前幸造，マイコンBASIC Magazine，2月号，139p.

▼FULO SUBETTELE
風呂場の石鹸に乗ってどこまで滑れるかを競う受験生には恐怖のゲーム。——速水PERSIA，マイコンBASIC Magazine，2月号，140p.

MZ-1500

▼移植版GOKIVADER
逃げ帰ろうとするゴキブリを上空で阻止するゲーム。——ピボ，マイコンBASIC Magazine，2月号，141p.

MZ-80B/2000/2500/2800

MZ-80B/2000/2200/2500

▼THE TAEKWONDO
格闘技ゲーム初のテコンドーゲーム。——ITO.S，マイコンBASIC Magazine，2月号，142-144pp.

MZ-2000/2200/2500

▼よかまちNAGASAKI
修学旅行で長崎に来た少年たちがカステラを集めるというゲーム。——森田敬太，マイコンBASIC Magazine，2月号，145-147pp.

MZ-2500

▼誌上公開質問状
MZ-2500でプログラムリスト中に文字コードを入力する方法などを解説。——編集部，マイコンBASIC Magazine，2月号，71p.

▼The five numbers　～五つの数字たち～
2人用対戦スゴロクゲーム。——河野　誠，マイコンBASIC Magazine，2月号，148-149pp.

▼NEW SOFT
麻雀パイを使ったパズルゲーム，ドラゴンを紹介。——編集部，LOGIN，1・2号，31p.

X1/X1turbo/Z

X1シリーズ

▼Might and Magic Book Ⅱ
難易度の高さで好評のM＆MⅡを，舞台の歴史，パーティの組み方などから徹底解剖。——編集部，テクノポリス，2月号，6-11pp.

▼特集こだわりレポート
Might and Magic Book IIを解説している。——エドモンドU，POPCOM，2月号，85-87pp.

▼なんでもQ&Aスペシャル　X1/X1turbo/X68000シリーズ編
ランゲージマスター（X1用CP/M）でカナを使用する方法などについて。——シャープ，マイコン，2月号，160-161pp.

▼水戸黄門88
BASIC＋マシン語で作られた，懐かしいタイプのAVG。さあ，君も助さん角さんとともに悪代官を倒せ！——TAMAちゃん，マイコン，3月号，274-286pp.

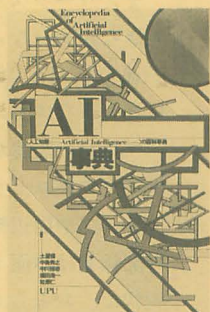
▼はらっぱ大相撲
MSX移植版相撲ゲーム。——ラムター，マイコンBASIC Magazine，2月号，188-189pp.

▼SANJA WARRIORS
殺人マシンSANJAを操って敵を倒すゲーム。——ズオ，マイコンBASIC Magazine，2月号，190-192pp.

▼NINJA WARRIORS—Kunoichiのテーマ；Are you…lady?—ゲームミュージックプログラム。——Yasuyuki，マイコンBASIC Magazine，2月号，214-217pp.

▼最新ゲーム徹底解剖!!
Might and Magic Book IIと新九玉伝，今夜も朝までPOWERFULまあじゃん2を研究している。——編集部，LOGIN，1・2号，174-177/186-189/198-201pp.

X1turboシリーズ

新刊書案内

本書は，あのGSを出版しているUPUから出た，戸田ツトムばりの装丁を持つ7,800円もする本です。AIに対して世間におもねたりせず，かといって専門家がいばったり，どこかの流れに偏ったりしない正攻法で挑んでいるのがポイントでしょう。AIというのは極めて間口が広く，かつ計算機学者だけでは成り立たない世界ですから本書の内容も多岐にわたっています。執筆者も計164人。目次を見ただけでも，ティーブレイク的なSFの話も含めて18章に分類されており，それぞれの章にいくつもの項目が並んでいるのがわかります。分類の仕方も，学習・言語理解・知識表現・パズル・知識工学といったAIらしいものから，こころ・知覚・脳など人間中心の話，情報・システム・言語といった計算機寄りの話，歴史・人物の背景的な話と豊富で，しかも各項目が独立しており，関連事項のインデックスまでついた文字通りどこからどう読んでもいい構成。「AI事典」というより副題の「人工知能の百科事典」というほうがしっくりするくらいです。AIに興味のある人や情報工学科の学生でなくとも，格好いい装丁と内容の豊富さから座右の1冊に欲しくなるでしょう。（K）

AI事典　ユーピーユー刊　☎03(222)1501
B5判変形　532ページ　7,800円

▼サイオブレード
アニメーションが話題のゲーム，サイオブレード。その売り（?）ともいえるメロディモジュールの使い道などを解説。——編集部，テクノポリス，2月号，17p.
▼特集こだわりレポート
新ゲーム，ピラミッドソーサリアンを解説している。——J・D・加藤，POPCOM，2月号，88-91pp.
▼特集こだわりレポート
新ゲーム，新九玉伝を紹介。——ウーやん，POPCOM，2月号，110-111pp.
▼FINAL-X
4人同時プレイ対応のレーシングゲーム。——星合健二，マイコンBASIC Magazine，2月号，193-194pp.
▼ハフマン圧縮プログラム
同誌'87年10月号からturboCP/Mに移植されたハフマン符号化を用いたデータ圧縮プログラム。48Kバイトまでのデータの圧縮が行える。CP/M用のマクロアセンブラM80が必要。——倶楽部X1 OVERFLOW，I/O，2月号，193-203pp.
▼なんでもQ&Aスペシャル X1/X1turboシリーズ編
CZ-8PK2とX1turboを使って倍角漢字を印字する方法や，X1turboシリーズでモデムターミナルを使用して受信した内容をプリントアウトする方法などについて。——シャープ，マイコン，2月号，160-162pp.

X68000

▼マウスでFM音源エディットを！
X-BASICで書かれたFM音源用の音色エディットユーティリティ。これでエディットした音色データは MUSIC PRO-68Kで使える。もちろんこのプログラムはCコンパイラでコンパイル可能。——石本淳，I/O，2月号，105-112pp.
▼高速スクロール・ドライバ
ED.Xを子プロセスから起動してスクロールを高速化するためのドライバ。アセンブラのソースリストで書かれているので福袋かあるいはXC付属のAS.Xが必要。——牧野和彦，I/O，2月号，123-126pp.
▼GAME BOX
サンダーフォースIIのグラフィックステージの全紹介。——編集部，I/O，2月号，130-131pp.
▼DIRCOPY
Human68k用のユーティリティコマンドでディレクトリ単位に簡単にコピーをするためのコマンド。AS.Xが必要。——市原昌文，I/O，2月号，186-191pp.
▼環境ソフト アメーバ
テキストVRAMのソフトキーボードのエリアにアメーバを増殖させ観察する環境プログラム。要AS.X。——M彦，I/O，2月号，244-245pp.

▼炎の猫
新世代プログラムBack Ground Pictureというこれは，グラフィックを大量に使った恐怖のおちゃらけプログラム。——唐笠じいさん，I/O，3月号，254p.
▼X68K Information Shop
X68000シリーズ用マルチタスクOS，OS-9/X68000とOS-9上でネットワークシステムを構成するためのハード付きソフトウェア，OS-9/NETの概略。——編集部，ASCII，2月号，289-290pp.
▼X68K Programmer's Shop
同誌1988年12月号のソフトウェアライブラリで紹介したパターンエディタPEで使用したオーバーラップ可能なマルチウィンドウシステムを採用するに至った経緯と，そのアルゴリズムについての紹介。——宮本親一郎，ASCII，2月号，291-293pp.
▼X68K Technical Shop
OS-9の特徴であるファイル管理システムやモジュールのリンクについて。——中山進，ASCII，2月号，294-295pp.
▼X68K Report Shop
ハンディプリントジャックとNEW Print Shop用グラフィックライブラリVol.1，2の価格と内容について。——編集部，ASCII，2月号，296p.
▼GAMING WORLD
最新ゲームのパワーリーグ，ガルフォース・怒濤のカオス，第4のユニット1・2，ボスコニアンを紹介。——編集部，テクノポリス，2月号，28-32p.
▼シミュレーションゲーム大特集
大戦略シリーズの最新作，SUPER大戦略68Kをレポート。——編集部，POPCOM，2月号，18-21pp.
▼X68000ワールド
殺人倶楽部，NEW Print Shop PRO-68K，めぞん一刻完結編，MIDIボード，Musicstudio PRO-68Kを紹介。——編集部，POPCOM，2月号，112-115pp.
▼なんでもQ&Aスペシャル X1/X1turbo/X68000シリーズ編
ハードディスクからワープロを立ち上げて自動的に辞書をRAMディスクに転送させるバッチファイルやHuman68k上でエスケープシーケンスを入力する方法の紹介，Musicstudio PRO-68Kの製品概要と価格などについて。——シャープ，マイコン，2月号，162-168pp.
▼Y-COM HotNews これがウワサのX68000版アフターバーナーだあ!!
X68000シリーズ用に発表される3Dアクションゲーム・アフターバーナーの開発中画面写真，アーケード版の説明など。——編集部，マイコン，2月号，249p.
▼X68000マシン語入門
スーパーバイザモードに関係の深い特権命令などについて。——高橋雄一，マイコン，2月号，367-377pp.

▼BATTLE CARNIVAL
トリガーを離したときにレーザーを発射する全11面宇宙戦闘機ゲーム。——山口和典，マイコンBASIC Magazine，2月号，195-197pp.
▼チャレンジ！X68000
アフターバーナーのサンプル画面，カサブランカに愛を，パックマニア，ザ・キングオブシカゴの紹介。——川野俊充，マイコンBASIC Magazine，2月号，292-293pp.
▼NEW SOFT
発売予定のミッド・ガルツを紹介。——編集部，LOGIN，1・2号，28p.
▼NEW SOFT
ガルフォース・怒濤のカオスを紹介。——編集部，LOGIN，1・2号，32p.
▼最新ゲーム徹底解剖!!
今夜も朝までPOWERRFULまあじゃん2と発売予定のMight and Magic Book IIを研究している。——編集部，LOGIN，1・2号，174-177/198-201pp.
▼X68000新聞
ヒストリーオブエルスリード，第4のユニット，めぞん一刻完結編，ガルフォース・怒濤のカオス，三國志，ラスト・ハルマゲドン，ソフトでハードな物語2を紹介。C-TRACE講座では色の付け方を解説。——編集部，LOGIN，1・2号，254-259pp.

ポケコン

PC-1245
▼DOT DRAGON
ドット単位のスクロールゲーム。——JG1TGS，マイコンBASIC Magazine，2月号，200p.
PC-1417G
▼誌上公開質問状
PC-1417Gでオリジナルキャラクターをスクロールさせる方法を解説。——編集部，マイコンBASIC Magazine，2月号，70-71pp.
PC-1500
▼THE万馬券
プリンタを使用する競馬シミュレーションゲーム。実在の馬のデータを分析している。——前田純男，マイコンBASIC Magazine，2月号，201-202pp.
PC-E500
▼スーパーゴルフ18H
2Dゴルフゲーム。コースは18ホール，スライス傾向からハンデまできめられるリアルなゲーム。——K.T.STARLET，I/O，2月号，173-175pp.
▼表計算プログラム
オールBASICの表計算ソフトの紹介。このプログラムは統計計算機能で入力したデータを集計するもの。——塚田洋一，マイコン，2月号，388—391pp.

デカルトの夢
テクノロジーの発展が人間や社会にどう影響していくか，という問題を考察する多くの著作の，これもひとつである。あらゆる知的活動が論理的演算によって処理できる世界。4世紀近く昔にデカルトの見たこの夢が，コンピュータという機構を得た現在，どれほど現実となりまた人間はそのなかでどう生きているのだろうか。2人の著者は，堅くなりがちなこうした問題を，数学者の立場から考察していて興味深い。
P.J.デービス，R.ヘルシュ共著　アスキー刊
A5判 344ページ 2,800円　☎03(486)7111

コンピュータ近未来学
どちらかといえばビジネスマン向けに，ハイテクノロジーのさまざまな話題をわかりやすく解説したものが本書である。超伝導にオプティカルディスク，人工知能にバイオ，ニュートリノに重力波など，ひととおりのポイントが詰まっていて，また内容も読みやすく，科学ジャーナリストとしてのこの著者らしい。こうした類の本は多く出版されているが，先端技術を駆け足で概観してみるのもなかなか楽しい。
那野比古著　日刊工業新聞社刊
B6判 256ページ 1,200円　☎03(222)7111

FROM READERS TO THE EDITOR

比較的暖かくて，冬だか春だかよくわからない今日この頃。今年はそれなのに風邪が流行しているらしいので，皆さん，健康には気をつけてくださいね。受験生諸君はもうひとがんばり，春はもうすぐそこまで来ています。

◆僕だけですね，「編集室へのメッセージ」と「STUDIO X」に載っているメッセージの区別がつかないのは。教えてください。「STUDIO X」に載るのには，このハガキに書けばいいのでしょうか。　光田　義広（14）愛知県

このコーナーには，Oh!X関係の話からプロレスや映画の話までなんでもかんでも載っているから，光田君が戸惑ってしまうのも，もっともかもしれませんね。毎月Oh!Xに付いている愛読者カードのメッセージ欄にそのまま書いてくれれば，こうして掲載されることになるのです。来月からもよろしくね。

◆1月号の「古きよき時代の工作少年」を読んで，自分にもこんな時代があったなぁ……と，懐かしく思いました。読んでいた雑誌も「初歩のラジオ」から「無線と実験」へと変わり，「トランジスタ技術」は最初のころから読んで（見て？）いましたが，最近ではもの凄い広告量でとんでもなく重い雑誌となってしまいました。
高橋　武志（35）富山県

◆1月号の特集を読んで，10年前にコンピュータを作っていた頃を思い出しました。なにしろまともなマイコン（パソコンではない）は皆無で，CPUとメモリとTTLを買ってきてはハンダゴテでリード線をゴチャゴチャ付け，ビデオ入力なんてないからテレビを改造してつなげてやっと動いていたものです。コンピュータ誌はハードの話しかありませんでしたから，今回の特集は非常に懐かしく思いました。しかし，あのような時代のハードの扱いからスタートすると，いまのハードのレベルに追いつくのには，きっと今月のような特集を3年くらい続けなければならないのでしょうね。木下　篤哉（26）愛知県

「マイコン」なんて呼ばれていた時代のハード講座と同じものをいまやろうとすると，3年やそこら続けたとしても，またハードのほうがずっと先のほうへ行ってしまって，きっといつまでも追いつくことは不可能なままでしょうね。

◆1月号の89ページ，特集の扉ページの写真はもしかしてあの懐かしい学研の電子ブロックじゃ，あーりませんか？
藤井　哲也（19）愛知県

◆究極のハードといえば，やはり学研の「電子ブロック」ですよね。あの乳白色のプラスチックケースのなかに入っていたメタルカンタイプのゲルマチューナー，ゲルマダイオード，巨大な電解コンデンサー，「黒点付き抵抗」などというわけのわからない呼び名。そのどれもが私の好奇心をツンツンしてくれたのです。当時，私は小学生でした。うっう，年がバレる。そーいや，電子ボードなんていうのもあったっけ。
柳井　敏彦（30）愛媛県

「電子ブロック」，「GIジョー」そして必殺の「魚雷戦ゲーム」……，ホントにいい時代でした。あの頃は誰もが柳井さんのように，あの電子ブロックのラジオから聞こえてきた雑音だらけの音に感激していたものです。いかんいかん，こっちも年がバレてしまう。

◆ハードウェアの特集で，少しでも多くの人がハードに興味を持ってくれることを期待しています。マシン語の知識さえあればデジタル回路なんて「へ」でもないので，すぐにストロングタイプになれるはずです。ぜひ，ハードで自分を磨いてほしいと思います。
小谷　史樹（19）北海道

◆ハンダゴテを握って10年以上になるが，この前，クルクルタコメーターというキットを原付に付けようとしたんです。そうしたら＋と－を間違えてICが溶けちゃったのだ。せっかくケースもバッチリ決めたのに。さっそくICを取り換えて作り直すつもりの私であった。
龍尾　謙二（22）岐阜県

◆毎月，18日が来るのが待ち遠しいくらい楽しく読んでいます。まだ，パソコンのことがよくわかっていませんので，こういう使い方ができるのか，こういうこともできるのか，と毎回新しい発見ばかりです。平井　康夫（40）大阪府

比較的とっつきづらいイメージを持たれているハード関係の記事でも，こうしてみんなワイワイガヤガヤ楽しんでくれているようです。だから，平井さんもハードとソフトの両面から時間をかけてゆっくりとご自分のものにしていってください。

◆「セサミストリート」は凄いと思う。特に出てくるモンスター（？）の人形たちである。手とか口の部分しか動かない奴もいるけど，実に生き生きとしていてどれも愛敬がある。某3チャンネルの教育番組のなかには，ほかにもたくさん人形は出てくるが，あのモンスターたちには遠く及びもしない。さすがはエンターテイメントのお国がらである。とにかく凝っている。だから「ダーク・クリスタル」のような，人間の出てこない映画も作れるのだろう。私のお気に入りのモンスターは，クッキー・モンスターとグローバーである。皆さんはどれがお気に入りなのでしょうか。郷司　謙一（19）神奈川県

セサミトストリートのキャラクターたちの凄いところは，1人ひとり（1匹？）の個性がその動きと同時によく表現されているところにあるんでしょうね，きっと。そういえば，この前は日本と中国で取材したのが放映されたようですが，私は残念ながら見逃してしまいました。郷司君は見たのかな？

◆（で）さん，「イースオタッキー」とか書いてありましたが，それをいうなら「イースオタク」

◀渡辺　久志　千葉県
アットホームな雰囲気がイラスト全体から伝わってくるようです。きっと渡辺さんはお子さんがかわいくて仕方がないんでしょうね。

◀磯村　賢二　東京都
なかなかに楽しい3コママンガ。しかし，このコマ割りのテクの背後に，某雑誌の気配を感じてしまったのは私だけではないはず。

大野　真実　静岡県
いいなぁ，こういうファンタジックな世界って。これだと，クリスマスカードも兼用で使えてしまいそう。

杉浦　豊　静岡県
そ，Oh!Xには仲間がいっぱい，だから安心してお読みください……，てなわけないでしょ。とにかくがんばって。

または「イースオタッカー」のほうが正解ではないでしょうか。なぜなら，「オタッキー」というのは状態を示す形容詞であり，人を表すのは「オタク」とか「オタッカー」なのです。僕はもと『LOGIN』の読者なので，このあたりのことはハッキリさせておかないと気になって仕方がありません。それと，(で)さんはスタッフのなかではひと際浮いてしまっているのでもう真人間には戻れないことでしょう。だからこのままオタクの道を極めるしか，残された道はないと思いますよ。　　山本　伸明（17）北海道

オタクの活用って，いったい誰が定めたんでしょうね。今度，その語源も教えてほしいものです。しかし，今月の特集のなかで(で)氏は「身から出たオタッキー」なる名言を残しています。果たして，時代の流れとともにオタキストはその筋を超えられるか，大いに注目していてください（なんちゅうコメントじゃ，これは）。

◆先日，障子を張り換えていて，ロール状の新品障子紙を買ったところ，「水に溶かすと糊になる」という化学糊の小袋が入っていた。実際にやってみると，180ccも糊ができてしまったんです。これには驚きました。そういえば，障子のない家も多くなってきているのではないでしょうか。　　千葉　生樹（17）宮城県

そういえば最近，障子の張り換えなんて見なくなりましたね。うちの実家なんて昔は旅館だったものだから，子供の頃なんか毎年年末になると，さあやるぞって大騒ぎしていたものです。

◆どうしてOh!Xにメガドライブのことが載っているの？　確かにX68000と同じ68000なんだけど……。いや，よく考えたらこれでいいのだ，Oh!Xとはこういう雑誌なのであった。
　　愛沢　太郎（20）東京都

うーん，こうも簡単にあっさりと納得されてしまうと，かえってこちらが悩んでしまいそう……。

◆ステゴちゃんのプレゼントをどーもありがとうございます。とってもカワイイですね。ステゴちゃんとステゴちゃんの入ってきたソフトバンクの段ボール箱は，宝物として一生大切にしたいと思います。　　牧　保志（15）熊本県

段ボール箱はどうでもいいけど，ステゴちゃんはかわいがってあげてください。あの号のプレゼントのなかでは一番の人気者だったんですから。

◆1月号166ページの尾崎さんへ。似たようなプロットの作品で，ジェイムズ・P・ホーガンの『未来からのホットライン』というのがあります。この本も，近未来の理論や技術を本当にあるように思えるほどリアルに描いています。タイムパラドックスをあれこれ苦心しながら核心へと迫っていく場面や，核融合炉の描写などは圧巻です。また読み物としても読みやすく，グイグイと作品の世界へと引き込まれてしまいます。そして最後のハッピーエンドでは誰もがホッとした気分になることでしょう。
　　赤城　豊和（21）神奈川県

◆ゲームの「めぞん一刻」の紹介記事を読んでいて思い出したのですが，テレビの「めぞん一刻」では，最初の頃，五代君が帰省したときのお土産は，石川県山中温泉にある石川屋さんの登録商標である「娘々饅頭（にゃぁにゃぁまんじゅう)」だったのに，最終回では五代君の実家は新潟になっていました。僕はマニアではありませんが，地元民としては放ってはおけません。誰か詳しい方教えてください。
　　篠田　和良（17）石川県

そうでしたっけ？　あんまりよく覚えていないなぁ。誰かこのお饅頭の謎を解明してください。

◆私はとってもせっかちです。新しいゲームを買ってきたときも，マニュアルを読まずにすぐ始めてしまいます。だから「スタークルーザー」も会話をどんどん飛ばしまくっています。そうすると当然，次の行き先がまったくわからなくなってしまいます。ついこの間，よせばいいのに会話を飛ばして，そのうえセーブまでしてしまったのです。だから，誰か心の優しいOh!X読者の方，惑星ハリケーンの次はどこに行ったらいいのか教えてください。
　　板垣　一彦（17）北海道

マニュアルを読まないのはよくある話だけど，スタークルーザーの場合は半分以上はAVGですからね。会話を飛ばしちゃったらただの迷子も仕方のないところ。誰か教えてあげてね。

◆12月号の「THE SOFTOUCH」を見て，X68000の「サンダーフォースII」を買いました。そして1月号の記事も読んで研究したんですけど，なかなかステージ1をクリアできません。こりゃ，難しい。記事にあった「ほんの小手調べ」という文字を見てボーゼンとしています。このゲームは女の子には向いていないのでしょうか？　でもステージ6を楽しみにチョコチョコやっています。でも，いまは受験生なので，合格したら（ムリかなぁー）思いっきりやろうと思っています。　　岡　幸子（18）山口県

ゲーム慣れしているうちのスタッフのなかでも，人間技ではクリアできないといわれているサンダーフォースII。でもステージ6は凄く楽しめるからなんとかガンバッて挑戦してみてください。でも，その前に受験をクリアしてからにしてね。

◆1月号に大阪のCMの話が載っていたが，某中○胃腸薬のCMもなかなかのものである。きれいな男性コーラスで「さわーやーかなー，い

第4回「言わせてくれなくちゃだワ」開催のお知らせ

皆さん，お待たせしました。いよいよ第4回「言わせてくれなくちゃだワ」を，来たる5月号において開催します。スタートしてもう4年目を迎えるこの企画。あれも言いたい，これも言いたい，皆さんが普段から心に思っていることのすべてを今月の愛読者カード（官製ハガキや封書でも可）にぶつけてOh! X編集室までお送りください。それらを一挙にまとめて大公開。これまでのようにイラスト，仲間募集，会報紹介などでも大いに結構。ジャンルは一切問いません。内容についてはなんでもありの無制限デスマッチ。スペースも存分に用意しておきます。ただし，合言葉はこれまでどおり「私が主役だ！」でいきましょう。

とにかく読者の皆さんの声，声，声の嵐を迫力十分にお届けする予定です。締め切りは3月10日到着分まで。今年こそ，かねてから念願の1000人掲載が果たして実現できるか!?　Oh! X編集室も目標目指して，スタッフ総動員でガンバります。さあ，ふるってご参加ください。皆さんからの元気いっぱいのお便りたくさん待ってまーす!!

◀高橋　哲史（19）福岡県
うんうん、受験生の感情がよくイラストに表われていてよろしい……、なんて講評されてる場合じゃないでしょ。勉強しなさいって、君は。

一ねーんまーく」と歌っているのだ。「さわやかな胃粘膜」とはいったいどのようなものであろうか。想像するだけで気持ち悪い。
川村　隆行（18）埼玉県
これは，しょっちゅう胸やけや二日酔いしている人にだけわかるキーワードなのかもね。

◆表紙の紙が気持ちＥぜぃ！　ザラザラしていてもう一最高(ぼかぁ変態じゃないぞ)。密かにOh!Xのロゴの下に「オー！エックス」とあるのもいい。先月，ハガキを出せなかった分いっぱい書いてる。と，思ったもののこれくらいしかネタがない……。　深津　厚二（17）島根県

◆今年はヘビ年だそうです。ということで「Oh!Xはドラゴンだ!」から，「Oh!Xはスネークだ!」とキャッチフレーズも一新して，初心者にもわかりやすい記事をよろしく。
宮崎　和也（18）佐賀県
宮崎君のおっしゃることはよくわかります。でも，年明け早々の編集室での（で）氏や荻窪氏たちの会話を聞いていると，どうやら今年は「Oh!Xはスネークマンショウだ!」，これでキマリみたいです。

◆本誌以外にお読みのパソコン雑誌：「赤本と参考書と各種問題集」。だぁー，浪人に正月はねぇ!!　大津　和之（19）福岡県

◆あと１カ月ちょっとである。なにがって？もち，入試に決まっているだろ。えっ？　20歳でローニン。悪かったなー，２浪で。ところで今年は８校も受けるので，な，なんと受験料だけで35万円もかかってしまう。医学部だからしょうがないけど，これだけあれば，Ｘ68000が買えてしまうというのに。ちなみに，一番高いところは５万５千円もするのだ。
安藤　憲興（20）東京都
ひぇー，受験料だけで５万５千円。1,000人受けたとしてそれだけで５千500万円。当然，そのあと入学金も入ってくるわけでしょ。やめられまへんなぁ，大学株式会社って。まっ，それはどうでもいいけど，安藤さんもがんばって合格してね。

◆アーケード版では500kmのスピードが出るフルスロットル。Ｘ68000版ではどうやら2000km以上出るようです。でも1000km以上になると「：」などの記号が表示されてしまいます。ステージ１～２でニトロを６連射すると，ステージ１で1100km以上，ステージ２では1900km以上になります。こうなってしまえばもはや車ではありませんが，皆さんも2000kmのスピードに挑戦してみてはいかがでしょうか。連続ニトロはギアをLOWにしてお使いください。
井上　雅夫（20）大阪府
こうなると，まるでジェット戦闘機なみですね。そのうちパラシュートをアイテムとして途中で拾わないと止まれなくなったりして……。

◆私のオーディオシステムは，未だにオール真空管です。音はいいんですが，冬になってコタツを使いながらスイッチを入れて音楽を楽しみ，そこからさらに冷蔵庫の電源をON。なんとブレーカーが飛んでしまうんです。借家だから文句もいえず，容量のアップもできず，当然パソコンの電源なんて入れられない日々です。
久保田　益志（25）長野県
パソコンと冷蔵庫の電源が同時に入れられないっていうのは悲惨な話ですね。でも，真空管のオーディオシステムを持っているなんて，うらやましい限りです。

◆フフフ，今年の正月は室戸岬までバイクで走ったぞー。完全武装で朝６時に出発。７時にフェリーに乗って，８時過ぎから徳島，阿南，東洋町を通って室戸岬へ。ざっと300km。途中，休みながら午後２時頃には測候所のわきを走っていた。暖かいのなんのって，もうバカ陽気。その日は民宿に泊まって次の日，高知市，大歩危，小歩危，金比羅，瀬戸大橋と走って，午後２時過ぎには家に着いていた。とても楽しかった。それにしても瀬戸大橋の通行料金は高い。バイクは通常料金の３倍以上取られる。マイカーは駐車場にいっぱい止まっているのに，バイクはせいぜい５，６台。そりゃそうだよね，金欠ライダーの来るようなところじゃねぇもんな。
寺尾　文治（37）岡山県
ほんとに四国は暖かい正月でしたね。ところで，寺尾さんがバイクで四国を走っていたとき，サンルーフを全開にして，「暑い，暑い」と騒ぎながらもスキー板３本乗っけて，高速道路を嬉しそうに走り回っていたゴルフGTIはいませんでしたか？　なにを隠そう，運転していたのはこの私です。

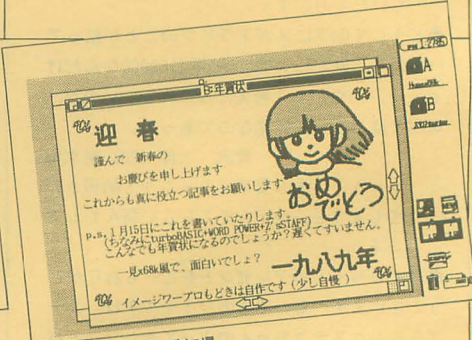
◆よく，「Ｘ68000ユーザーは購買力がある」なんていわれてるけど，ただ単に，貯金もできずに苦しんでいるだけなんだぞ。
小林　徹（19）茨城県

◆昨年５月号の「言わせてくれなくちゃだワ」で，「X1turbo ZIIを買うぞ」と載っていたのだが，Ｘ68000ACEの出現でＸ68000ユーザーになってしまったおかげで，友人から「う・ら・ぎ・り・も・の一」と呼ばれています。嬉しいやら，悲しいやら……。　釈永　聡（17）富山県
今回またこうして載ってしまえば，明日からは正々堂々と「私がＸ68000ユーザーの釈永だぁ」と胸を張っていえますよね，きっと。

◆あの富士通の32ビット・ホビーマシンの広告のコピーに，「平成パソコン」なんて使われ始めると手強いかもしれない。
竹谷　直樹（20）静岡県
「国民機」に「平成パソコン」なんてことになったら，やっぱりシャープさんの切り札は「世界初」しか残されていないのでしょうか。

◆Yumiさんに遅れること３年半。僕もようやくFZ-250を買いました。Yumiさん，今度僕と一緒にツーリングしましょう。
佐藤　克美（20）埼玉県
Yumiさんは霧降り高原で忙しくされているようなので，ツーリングに誘うのはちょっと無理かもしれません。代わりに，霜降り高原の（で）氏でも今度誘ってやってください。

◆パソコンゲーム界のレコード大賞ともいうべき，GAME OF THE YEARの頂点に立つのはいったいどのゲームなんでしょうか。それは４月号の発表を待つとして，さて，僕は去年のレコード大賞を見ていたのですが，こともあろうに光GENJIが大賞をとってしまったのだ。あれは歌がうまいわけではなく，ただ単にレコードが売れただけではないか。ああ，レコ大も落ちぶれたものだぜい。しかーぁし，これは誰も悪くない。あくまでもデータに基づいた結果なのである。見る目を持たない大衆の目というものは恐ろしいものである。まあ，彼らもこのあとは

▲村田　敦司　神奈川県
こちらこそ今年もよろしく。ところで，村田君のイラスト見てて思い出したけど，ドラクエⅣっていつ発売になるんでしょうね。

▲大島　靖（21）愛知県
イメージワープロもどきが自作なんて凄いですね。少しどころか，ばんばん大いに自慢しちゃってくださいな。

フォーリーブスと同じ運命をたどることであろう。それから断わっておくが，私は受験生である。　　村上　隆広（18）石川県

フォーリーブスって，もう少し歌は聞けたような気がするけど，光GENJIのほうはねぇ……。それがレコード大賞をとってしまうとは意外や意外。紅白といい，レコード大賞といい，年末恒例のテレビ番組がますますつまらなくなってしまいました。でも，Oh!XのGAME OF THE YEARはそんなことは絶対にありませんから，期待してください。

ぼくらの掲示板

●掲載ご希望の方は，官製ハガキに項目(売る・買う・氏名・年齢・連絡方法……)を明記してお申し込みください。
●ソフトの売買，交換については，いっさい掲載できません。
●取り引きについては当編集室では責任を負いかねます。
●応募者多数の場合，掲載できない場合もあります。

仲　間

★「JMC1500」では，MZ-1500ユーザーを対象とした会員を募集します。活動は月1回発行の会報を中心に，誰もが楽しめるサークルを目標として頑張っています。現在，会員は13名。入会にあたっては年齢，性別，パソコン歴などは一切問いません。入会ご希望の方は60円切手同封のうえ封書にて連絡を。〒312 茨城県勝田市中根3271-85　照井貴之（18）

★「CUREC」ではX1のディスクユーザーを対象とした会員を募集します。特にFM音源を持っている方，大歓迎。活動内容は音楽プログラム，D&D，小説，ゲーム紹介などの内容の毎月20～28ページ（年2回40ページ）の会報発行のほか，年2回ディスクを会員に送付して優秀作品を選出するFM音源大会などを行っています。興味のある方は60円切手同封のうえ封書にて連絡を。また，60円切手3枚同封された方には会報を添えて送付します。〒488 愛知県尾張旭市三郷町陶栄97　山田亮介（17）

★X1turboユーザーを対象とした「X1たんぼの会」を発足させるにあたり，会員を募集します。当会の活動目的は，ソフト/ハードの情報交換をメインに考えていますが，ハード関係ではかなり詳しいメンバーが参加していますので，幅広いサポートが可能だと思います。興味のある方は60円切手同封のうえ封書にて連絡を。〒639-11 奈良県大和郡山市小林町中9-18　八木あきこ（22）

★僕はX1Gmodel 30を購入したばかりのユーザーです。ですからX1に詳しい方にいろいろと教えてほしいと思っています。経験豊富な方ご連絡ください。〒032 岩手県久慈市田屋町2-29-6　大粒来茂樹（14）

★クラブ「RX1」ではX1，PC-88シリーズのディスクユーザーを対象とした会員を募集します。活動内容はソフトの情報交換などです。詳しいことをお知りになりたい方は，60円切手同封のうえ封書にて連絡を。〒676 兵庫県高砂市米田町島12-1　永井幸志（18）

★「SPX」では，X1ユーザーを対象とした会員を募集します。主な活動内容は，ゲームの情報交換を主体とした会報を月1回発行しています。初心者やゲームファンは大歓迎。詳しいことは60円切手同封のうえ封書にて連絡を。〒260 千葉県千葉市真砂1-10-5　福田信昭（15）

★「X1た～ぼCLUB」では，X1turboユーザーで初心者の方を中心に会員を募集します。活動はゲームの情報交換とディスクによる会報発行です。会費は入会金100円，月会費200円です。詳しいことをお知りになりたい方は60円切手同封のうえ封書にて連絡を。〒862 熊本県熊本市湖東3-6-5　村井友治（18）

売ります

★MZ-2000用拡張I/OボックスMZ-1U01，プリンタインタフェイスMZ-1E08，プリンタケーブルMZ-1C32，プリンタMZ-1P07（マニュアル，リボン付き，箱なし）のセットを送料込み3万円で。連絡は往復ハガキで。〒501-42 岐阜県郡上郡八幡町市島1936　清水昭仁（29）

★ディスプレイMZ-1D22（付属品付き）を2万5千円で。また，NECのTV-351（マニュアル，ケーブル，リモコン付き）を4万円前後で。連絡は往復ハガキで。〒280 千葉県千葉市大森町319-13　黒沢利夫

★ヤマハのキーボードPSR-16（新品同様）を，1万8千～2万円で。またはカシオのCT-370との交換も可。連絡は往復ハガキで。〒680 鳥取県鳥取市湖山町北3-325学生寮「寿」　古川公彦

★カラービデオプリンタCZ-6PV1（箱，マニュアル，付属品付き，美品）を，送料込みで8万～10万円くらいで。連絡は希望価格明記のうえ往復ハガキで。〒232 神奈川県横浜市南区大岡5-10-14　鈴木征人（18）

★ブラザーのドットプリンタM-1009（リボン，X1用ケーブル，マニュアル，箱付き）を送料別1万1千円で。連絡は往復ハガキで。〒183 東京都府中市朝日町1-15-26　大葉利夫（38）

★MZ-700/1500用ドットプリンタMZ-1P14（新品同様）を送料込み2万円で。連絡は往復ハガキで。〒981 宮城県仙台市北根1-2-7　緑川健（19）

★データレコーダCZ-8RL1を1万円前後で。連絡は希望価格明記のうえ往復ハガキで。〒980 宮城県仙台市弓ノ町12　伊藤洋美（39）

★X1用FDD・CZ-503F（1年間使用，完動品，傷なし）を，送料込み1万7千円で。連絡は往復ハガキで。〒229 神奈川県相模原市西大沼2-30-7　宍戸高志（17）

★X1用FDD・CZ-503F（付属品付き，箱なし）を1万8千円で。またFM音源ボードCZ-8BS1（付属品付き，箱なし）を1万3千円で。各送料込み。連絡は往復ハガキで。〒700 岡山県岡山市津島東4-7-47生泉荘11号　西本聖（19）

★X1用FM音源ボードCZ-8BS1を1万円で。またMZ-2000用拡張ユニットMZ-1U01を5千円で。各箱，マニュアル付き。連絡は往復ハガキで。〒192-03 東京都八王子市南陽台2-7-11　須藤義一（20）

買います

★MZ-2500用増設VRAM/MZ-1R27（コンパチ品でも可）を，送料込み5千円以内で。連絡は価格明記のうえ往復ハガキで。〒331 埼玉県大宮市日進町1-585-4　今野和浩（18）

★X1用320KB外部RAMボードCZ-8EM，またはCZ-8BE2を1万円で。また，X1turboII用JIS第2水準漢字ROMを3千円で。各送料込み。連絡は往復ハガキで。〒270 千葉県松戸市常盤平双葉町19-30　森尾博昭（18）

★X1turboII用第2水準漢字ROM・CZ-8BK4を送料込み3千円で。連絡は状態を明記のうえ往復ハガキで。〒146 東京都大田区南久が原2-23-9　松尾勝（15）

★高性能CRTフィルタBF-68PROを，送料込み1万円以内で。また，プリンタCZ-8PC3（箱なし，傷可）を送料込み3万円前後で。連絡は，付属品の有無と希望価格を明記のうえ往復ハガキで。〒259-03 神奈川県足柄下郡湯河原町土肥4-5-24　佐々木貴志（17）

★X1用漢字ROM・CZ-8BK2を送料込み5千円で。連絡は往復ハガキで。〒815 福岡県福岡市南区桧原7-25-33　藤本剛太（17）

★X1用FM音源ボードCZ-BS1（付属品付き，箱はなくても可）を送料込み1万円で。連絡は希望価格明記のうえ往復ハガキで。〒511 三重県桑名市上深谷町1470-2　羽根田雅之（17）

★X1用漢字ROM・CZ-8BK2と拡張I/Oボックスのセットを8千円前後で。X1用モデムユニットCZ-8M1を8千円前後で。また，X1用FM音源ボードCZ-8BS1を8千円前後で。各送料込み。連絡は往復ハガキで。〒124 東京都葛飾区西新小岩1-1-2-816　遠井将一（18）

★X1turbo用シャープ純正400ラインディスプレイを2万5千～4万円くらいで（程度によって希望価格を決めてください)。TVチューナー付きはプラス1万円で。また，Oh!MZ1986年2月号を800円くらいで。連絡は電話番号と価格明記のうえハガキで。〒440 愛知県豊橋市多米町大門21-3　大木敬哲（14）

バックナンバー

★Oh!MZ1986年9月号，1987年3月号（切り抜き不可）を送料込み各1,000円で。連絡はハガキで。〒598大阪府泉佐野市大宮町12-31　矢倉利一（16）

★Oh!MZ 1986年12月号，1987年1，6～10月号を送料込み各1,000円で。ハガキ以外切り抜き不可。連絡は往復ハガキで。〒289-13 千葉県山武郡成東町成東2470　安井忍（20）

編集室から from E・D・I・T・O・R

DRIVE ON

このコーナーでは，本誌年間モニタの方々のご意見を紹介しています。今月は１月号の記事に関するレポートです。

●まったく日頃思っていることが「ハードウェアをめぐる冒険」に書いてありました。あの松ヤニの焦げる匂いに中毒になっている僕は，これまでいろいろな人にハードを勧めてきました。ソフトをやっている人にもハードの知識は必需だと思います。僕の場合，自分なりにですがCPUの働きなどを学んできたので，いろいろな面でコンピュータの機能を容易に理解することができました。機器に異常が起きたときでも，ハードの知識があれば簡単に解決できます。ハードとソフトに共通点が多いということは，ソフトの知識があればハードも理解しやすいということでしょう。僕は「ANDもORこわくない」を読んでTTL規格表を注文しましたが，これからは必携ですね。今回はD-FFが主に解説されていましたが，T-FFとかそのほかのフリップフロップにももっと言及してくれてもよかったんじゃないですか。また「大きなノイズの使い方」で，いきなり隙をついてくるのが栞野さんらしいと思います。普通ノイズといえばどこでも嫌われ，それを抑えるために皆四苦八苦していますよね。それを利用してしまおうというのがいい。

星　大地（15）MZ-731，CZ-611，PC-1475　静岡県

●堅苦しく面白味のない文章になりがちなハードウェアの内容ですが，「ハードウェアをめぐる冒険」はそれを気楽に読ませてくれるよい記事だったと思います。「ANDもOR もこわくない」や「BASICでわかる論理回路」では，Oh!Xがマニアの読む雑誌と言われるわけがわかりました。また，「512Kバイトの誘惑」は，X1turboユーザーにとってはたいへん実用的でまたわかりやすいものでした。多くのユーザーにとり「実用的」というのは大切なことだと思います。今後のサポートにも期待したいですね。

青木　民夫（33）PC-9801VX　富山県

●「512Kバイトの誘惑」。いやあNEW Z-BASICでもうれしかったのにRAMの拡張もこんなに簡単にできるなんて。実はＺが4096色対応になったとき，色数は増えてもメモリが足りない感じがして少し不安だったのですが，これで僕のＺも天下無敵！なはずですが，Ｚの特徴を生かしたソフトがない。残念なことです。「X68000CP/M-80システム」は，起動するにはX1＋X68000＋CP/Mとなかなかの環境が必要ではありますが，利用価値は十分にあるのでX68000ユーザーの方にはハード製作のよい機会にもなると思います。

渡辺　知己（16）X1turboZ　北海道

●パソコンで乱数を作るといえば，単にリフレッシュレジスタを加工するものから始まって，かなり奥の深い話になります。加減しかできないアセンブリ言語で乗除・指数・対数・無理数を扱うのは面倒だが，不可能ではない。なのに乱数は適当にビット処理したりタイマとからませたりして苦労して加工しても結局ダメだった，というのが少なくなくて，僕もゲーム作りのとき困ったものでした。「大きなノイズの使い方」は，それをハードで処理しようというわけですね。トランジスタのノイズを使うのは安上がりだし，コンパクトになって妥当だと思います。ところで，サンプルがBASICでひとつだけなのは寂しい。マシン語でちゃんとタイミングの取れるものも欲しかったです。

上野　壮也（17）MZ-1500　大阪府

●僕はハードとはさほど付き合いは深くないが，以前友人にアドレスデコードについて習ったことがある。すぐに理解できたが，そのときひとつ気づいたことがあった。それは，デジタルの知識だけではなにもできない，ということである。プログラミングでは，「目的」をすべて０，１で表す（プログラムする）ことができるが，ハードウェアの場合はアナログの知識が必要である。もともとコンピュータ自身アナログをデジタルとみなして動かしているので当然のことだが，そのアナログとデジタルのかみあわせといったことをもっとよく知りたくなった。

猪原　弘晃（18）X1turbo　兵庫県

●「X68000CP/M-80システム」ですが，「S-OSに次ぐ勢力を誇る(?)」8ビットOS，CP/M-80に関しては，やはりその歴史とアプリケーションの蓄積を無視できません。それが64180の10MHzノーウェイトで動くとなればCP/Mに慣れ親しんだ人にはこたえられないと思います。それから，毎年GAME OF THE YEARでOh!MZ賞が寂しくなっていくのは，時代の流れとはいえユーザーにとっては悲しいことです。そんな中で，古籏さんのスペハリや，ノミネート外ですがALANなど，読者が頑張っているのは唯一の救いといえるでしょう。実際に全部のゲームをプレイしたわけではありませんが，「個性」の強いゲームがもっと出てきてもいいのではないかと思います。自己主張を持ったゲームも確かにありますが，「新しい！」と思えるような光っているゲームは，海外の作品に多いのが事実です。日本のソフトハウス各社が新しい方向を摸索しているのはわかりますが，どうもドングリの背比べの感が拭いきれません。ひとつでも多くの「光った」作品が今年のGAME OF THE YEARでも選ばれることを期待しています。

今野　和浩（18）MZ-2521，PB-100，FX-780P，PC-E200　埼玉県

●過去すべてのGAME OF THE YEARをみてきた僕としては，投票する人々の間で，年ごとにあるひとつのゲームに対する思い入れというものが減ってきているような気がします。これはやはりゲームが多様化・複雑化してきたからなのでしょうか。最近はたくさんのゲームを結構自由にプレイできるようになってきたことも，その理由のひとつかもしれません。今年はおおっ！　とうなるような強烈な投票が増えるといいな，と期待しています。それからHyper Game Book。やはりMZユーザーはすごいと感じました。

橋本　浩二（17）X1F model 30，CZ-611　兵庫県

ごめんなさいのコーナー

1989年２月号　Daddy Mulk

P.154　このプログラムをX-BASIC V2.0で実行する際には，あらかじめCONFIG.SYSを，

DEVICE＝OPMDRV.X　#80

のように設定しておいてください。なお，正しいタイトルは「Daddy Mulk」です。

1989年１月号　FLICK

P.78　キャラクタの一部で標準以外のものが使われていました。

5010H　A5 A5 A5 A5→2E 2E 2E 2E

に変更してください。

1989年１月号　マシン語ゲーム工房

P.66　X1/X1turbo用のプログラムで一部説明が抜けていました。PCGのデータ部分はMZ-2500用とまったく同じですので，リスト４にはリスト3の191行以降をくっつけてからアセンブルしてください。

1986年７月号　FM音源ミュージックシステム

P.118　ALL#，n～の命令で一部誤動作がありました。

B539H　10　→　11

に変更してください。

バグに関するお問い合わせは
☎03(263)2230(直通)
月～金曜日16：00～18：00

お問い合わせは原則として，本誌のバグ情報のみに限らせていただきます。入力法，操作法などはマニュアルをよくお読みください。
また，よくアドベンチャーゲームの解答を求めるお電話をいただきますが，本誌ではいっさいお答えできません。ご了承ください。

BASICです遊んでいただきますきゃ♡

▼春一番は，なんでもござれのBASIC特集です。ゲームありグラフィックあり，言語解析の話に引力の話あり，またオタク族も出てきてさぞ楽しんでいただけたことでしょう。手軽に使えるBASICで，気軽にいろいろ試してみる。そうして何か面白いことができたら，ぜひ編集室までお知らせください。

▼いよいよ来月はGAME OF THE YEARの発表です。どの賞をどのゲームが獲得するか，どんな飛び入りオリジナル賞が出てくるか。思い切りにぎやかな特集にしますのでご期待のほどを。

▼1年のサイクルなんてあっという間です。ゲーム特集のあとには，かの「言わせてくれなくちゃだワ」や創刊7周年記念が控えています。また大々的に皆さんの声を募集しますのでよろしく。あ，メッセージだけでなくイラストとか投稿プログラムとかなんでも自由にどうぞ。今月は投稿募集のコーナーも新しくなりました。145ページを見てはりきって応募しましょう。

▼村田敏幸氏によるX68000マシン語プログラミングの連載は来月から始まります。皆さんの要望などをたくさんお寄せください。

▼今月から復帰する予定だったBetween The Linesの勝本信氏でしたが，スケジュールの都合で延期になってしまいました。すみません。もう少しお待ちください。

▼先般お知らせしたS-OS"SWORD"のコピーサービスですが，2月中旬より順次発送の予定です。また，コピーサービスはバックナンバーの入手できない機種に限りますので，在庫状況をご確認のうえ，お申し込みください。

▼暖冬の声しきりでしたが，さすがに2月に入ると寒くなりました。インフルエンザも昨年暮れからの猛威がなかなか衰えませんね。電車に乗るとマスク人間だらけで，風邪ビールスまでが公共物。仕事では時間に追われるばかりだし，やっぱり冬って暗い季節かなあ。

そういえば今月末のイントレランスのスケジュールはひどい。ウイークデーの夕刻からなんて，社会人をばかにするにもほどがある。文句を並べてしまってごめんなさい。皆さんも健康には十分気をつけましょう。ではまた来月。

投稿応募要領

- 原稿には，住所・氏名・年齢・職業・連絡先電話番号・機種・使用言語・必要な周辺機器・マイコン歴を明記してください。
- プログラムを投稿される方は，詳しい内容の説明，利用法，できればフローチャート，変数表，メモリマップ（マシン語の場合）に，参考文献を明記し，プログラムをセーブしたテープ（ディスケット）を添えてお送りください。また，掲載にあたっては，編集上の都合により加筆修正させていただくことがありますのでご了承ください。
- ハードの製作などを投稿される方は，詳しい内容の説明のほかに回路図，部品表，できれば実体配線図も添えてください。編集室で検討の上，製作したハードが必要な場合はご連絡いたします。
- 投稿者のモラルとして，他誌との二重投稿，他機種用プログラムを単に移植したものは固くお断りいたします。

あて先

〒102 東京都千代田区九段南2-3-26井関ビル
日本ソフトバンク出版部
Oh！X「㋢㋮㋮㋱」係

S H I F T · B R E A K

▶以前，自動車のCMにあった「くう，ねる，あそぶ」というコピー（あれ，今でもやってたっけ?），なかなかユニークだ。これを「食う，寝る，遊ぶ」というふうに漢字にしてしまうとぐうたら人間の単なる戯れ言になってしまう。わざと全部ひらがなにする糸井重里氏の才能は，やっぱり「ほんもの」ということかな。（R.K.）

▶先月書き忘れたこと。殺人倶楽部のレビューでは友人にかなり助けてもらいました。DAIVE君，ありがとね。ところで，OS-9のMW-Cが59.8K円。かつての6809版が160K円のころに比べれば安くなったのかもしれないが，MS-DOSでは10K円台のCは普通なんだよね……。標準で高級言語がつかないんだからもう少し何とかならないのかな？（で）

▶実をいうと遅ればせながら村上春樹にはまっている。これだけメジャーになっちゃうと読んでるのが恥ずかしいなぁとも思うけれど，一度読んじゃうとやめられませんなぁ。今年に入ってからもう20冊である。本代もばかにはならない。

お勧めは『世界の終りとハードボイルド・ワンダーランド』だね。（C.W.）

▶げげ，いつのまに。レヴュのアルバムが出てる！去年の秋だって？ そういや最近レコード買ってなかったもんな。げげ，新宿ロマンがなくなっちゃってるよ。飯田橋佳作座も消えてるじゃん。そういや最近は映画見てなかったもんな。池袋文芸座はまだ残ってるね？ 最後の砦だもん，ガムバってもらわなきゃ。あ，オチがない。（Mu）

▶最近気づいたのだが，私の知っている北海道人はみな寒がりである。東京で，しかも暖冬といわれているなか，厚着をして街を歩く。シャツ1枚で大丈夫な室内でも上着を脱がない。ある北海道出身者は，北海道では室内は汗ばむほど暑いから，といっていた。彼らには絶対的な気温より，外気との温度差，つまり相対温度のほうが重要らしい。（K）

▶最近いろいろな食べもの屋を探しては食べにいっている。会社を比較的早く（たとえば8時ごろ）出た日などは，1時間くらいのところまで足をのばし，新しい店のノレンをくぐる。しかし，いかにも独身男性会社員（しかも残業の多い）の行動といえる。こんな自由な毎日はそう長くは続かないだろう。続けようと思えばできるのだろうが。（K.S.）

▶4年間の寮生活におさらばして独り暮らしを始めた。いままでは2人部屋だったが，これからは寝ぼけた同室の人に顔をたたかれることはない。同室の人に気兼ねなく夜更かしできる。毎朝同室の人のセットしている2つの目覚まし時計で無理矢理起こされることもない。いままでよく我慢してきたものだ。（ユニットバスの使い方がわからないKO）

▶年賀葉書がとうとう売れ残ってしまったそうです。新聞には例の自粛の影響だなんて書いてありましたが，私がニランだところでは，本当の理由は2等の賞品がキャプテンの端末だったからだと思われます。あーゆーのを「有難迷惑」もしくは「不良在庫の整理」というのでしょう。ホントにもう，いいかげんに諦めたらどうなんです。（M）

▶CDVとでっかく書いてあるニュースレターをなにも考えずに書店で買ってきてなかを見たら犬科の動物のかかるジステンパービールス（Canine Distemper Virus＝CDV）がバルト海あたりのアザラシたちをたくさん犠牲にしている病気のビールスと非常によく似ているという記事だった。なんでみんな略語を混乱なく使えるんだろう。（よ）

▶「やっぱり16×16の128個じゃね」「でもBGは2枚ありますよ」「大阪ではもう予約注文とってるよ」「消費税はかかるんでしょ」「32ビットとしては破格だし」「キーボードは別売りですね」「ちゃんとプロテクトモードだし」「え！ ROGUEが動かないんですか？」ちまたはなにやら騒がしい今日この頃。（U）

▶先日，秋葉原にあるビルの地下にコーヒーの美味しい店を見つけた。そのあと外に出て会社に電話しようとしたら，電話ボックスはあってもなぜかシステム手帳を広げたサラリーマンにどこも占領されている。店頭には腐るほど電話機が並んでいるというのに，仕方がないからまた別の喫茶店に入ってしまった私は健全なサラリーマン？（N）

▶問題のUFOキャッチャーなんですが，私の見込んだとおりの大ヒット。ただ，新宿，渋谷，池袋と各地のゲーセンを回って気がついたことに，人形の質が渋谷だけ妙にいい。キャラクターデザインもいいし，バリエーションも豊富。客層が違うからかもしれないけど，私の目はごまかされないんだから。（100円で3つもヌイグルミを獲得したT）

microOdyssey

今月のC.W.氏の編集後記にもあるように，時代は村上春樹のようである。『ノルウェイの森』はベストセラーとなり，原色を使った大胆な装丁とともに，すっかり若者の間でファッションしている。おまけにレコードショップではビートルズコーナーが急遽拡大され，CDの便乗商法も大繁盛している。「若者の活字離れ」という言葉はもうひと昔のものであるが，いまの村上春樹現象からすると，現在においては「若者が活字を選ぶ時代」へと移行しつつあるようだ。

私は『ノルウェイの森』については別に新しいとは思わない。年がバレてしまうかもしれないが，あのノリは遠い昔に柴田翔や庄司薫，村上龍，さらにはサリンジャーなどで経験している。しかし，いまの世代に共感を与えるものであることも理解しているつもりでいる。いつだって，ある世代が感性を共有する瞬間は存在する。それを今風にうまく表現するかが作家という職業の腕の見せどころであり，また読む側はこれまで以上に読みやすく，感性を刺激してくれるものを選ぶ。

このような流れは，誰の目にも明白なものであろうと自分勝手に判断し，いまはいま，さて明日はどっちだ，などと鼻歌交じりに朝刊を開いた1月15日のこと。成人式当日，なにを思ったかその新聞には「社説を読まない若い世代へ」と題した社説が載っていた。

その内容はといえば，「成人式を迎える187万人の皆さん，社説を読みましょうキャンペーン」なのであった。そこには社説が読みづらい理由として，第1に「忙しくて時間がない」，第2に「文字がギッシリで難しそう」という理由が挙げられ，逆に社説を読むことのメリットとして「ニュースの奥行きと背景がつかめる」，「世の中の流れをつかむのに便利」，そしてなにより重要なのが「社説の主張により新聞の個性がわかり，新聞を読むことが面白くなる」のだそうだ。

新聞が個性を持つことは大いに結構なことなのだが，それを改めて社説を使って大声で訴える個性とは？　謎が謎を呼び，暗雲漂い，新聞料金の適正価格とは果たして存在するのか，とわけのわからん疑問まで発生する。

話は変わって，東京近郊で限定発売され人気を集めている女性週刊誌『Hanako』の「新聞を読まなくても，政治・経済の話ができるページ」というなが一いタイトルの連載ページ。1月26日号のタイトルは「パレスチナ独立国家宣言！日本の新聞を読んでもわからん。だから真実を書く」という，これまた女性誌とは思えないような乱暴なタイトルのものが載っている。

そこには「複雑でわかりづらい問題だから大胆にダイジェストにしてみます。きっと糸口が見つかるでしょう」とも書かれている。こうも正面切ってお題目をずらずら並べられると，私なんかは赤面してしまいそうだが，きっといまの若い女性を対象にした場合にはこれがひとつの方法だとも思う。

と，なると先の社説とはいったい？　読者に合わせた読みよい紙面を前提に作られているはずの新聞と，若いOLを対象とした軽いノリの新刊女性週刊誌。一方は今年でなんと110周年を迎えるという。これはどう考えても，Hanakoさんの一本勝ちである。　（N）

1989年4月号3月18日(土)発売

特集　激突!!　新作ゲームvs深夜族
1988年度"GAME OF THE YEAR"発表!
新連載　68000マシン語プログラミング入門
全機種共通システム　SLANG用実数演算ライブラリ
Oh!X　LIVE　in '89
X1/turbo用スキーム/X68000用パワードリフト
特別付録　X68000イメージポスター

バックナンバー常備店

東京	神保町	三省堂神田本店5F 03(233)3312
	〃	書泉ブックマートB1 03(294)0011
	〃	書泉グランデ5F 03(295)0011
	秋葉原	T-ZONE 7Fブックゾーン 03(257)2660
	八重洲	八重洲ブックセンター3F 03(281)1811
	新宿	紀伊国屋書店本店 03(354)0131
	高田馬場	未来堂書店 03(200)9185
	渋谷	大盛堂書店 03(463)0511
	池袋	西武百貨店11Fブックセンター 03(981)0111
	〃	西武百貨店9F コンピュータ・フォーラム 03(981)0111
	町田	久美堂東急ハンズ店 0427(28)2783
神奈川	横浜	有隣堂横浜駅西口店 045(311)6265
	〃	有隣堂ルミネ店 045(453)0811
	藤沢	有隣堂藤沢店 0466(26)1411
神奈川	厚木	有隣堂厚木店 0462(23)4111
	平塚	文教堂四の宮店 0463(54)2880
千葉	柏	新星堂カルチェ5 0471(64)8551
	船橋	西武百貨店10Fブックセンター 0474(25)0111
	〃	芳林堂書店津田沼店 0474(78)3737
	千葉	多田屋千葉セントラルプラザ店 0472(24)1333
埼玉	川越	黒田書店 0492(25)3138
	川口	岩渕書店 0482(52)2190
茨城	水戸	川又書店駅前店 0292(31)0102
大阪	北区	旭屋書店本店 06(313)1191
	都島区	駸々堂京橋店 06(353)2413
京都	中京区	オーム社書店 075(221)0280
愛知	名古屋	三省堂名古屋店 052(562)0077
	〃	パソコンΣ上前津店 052(251)8334
	刈谷	三洋堂書店刈谷店 0566(24)1134
長野	飯田	平安堂飯田店 0265(24)4545
北海道	室蘭	室蘭工業大学生協 0143(44)6060

定期購読のお知らせ

Oh!Xの定期購読をご希望の方は，最寄りの郵便局にある払込用紙に，

口座番号　東京1-29307
加入者名　株式会社日本ソフトバンク

とご記入のうえ，年間購読料6,500円を添えてお申し込みください。その際，裏面の通信欄に「○年○月号よりOh!X定期購読希望」と忘れずに明記してください。なお，すでに定期購読をご利用いただいている方には，購読期限終了と同時にご通知申し上げますので，同封の払込用紙をご利用ください。

海外送付ご希望の方へ

本誌の海外発送代理店，日本IPS(株)にお申し込みください。なお，購読料金は郵送方法，地域によって異なりますので，下記宛必ずお問い合わせください。

日本IPS株式会社
〒101 東京都千代田区飯田橋3-11-6
☎03(238)0700

3月号

■1989年3月1日発行　定価540円　■発行人　孫　正義　■編集人　笹口幸男
■発行元　(株)日本ソフトバンク
■出版事業部　〒102 東京都千代田区九段南2-3-26 井関ビル　☎03(261)4095　FAX 03(262)8397
編集室☎03(239)4156
出版営業☎03(261)4095
広告営業☎03(297)0181
■本　社　〒102 東京都千代田区九段南2-3-14　靖国九段南ビル　☎03(263)3690(代)
TELEX 東京　232-4614JSBTYJ　FAX 03(263)3660
■西日本営業部　〒541 大阪府大阪市東区南本町2-6　明治生命堺筋本町ビル10F
☎06(264)1471(代)　FAX 06(264)1481
■印　刷　凸版印刷株式会社

©1989 SOFTBANK CORP.　雑誌 02179-3　本誌からの無断転載を禁じます。

BACK ISSUES バックナンバー案内

ここには1988年3月号から1989年2月号までをご紹介しました。現在1997年4, 1988年1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 1989年1, 2月号までの在庫がございます。バックナンバーおよび定期講読のお申し込み方法については，本文166ページを参照してください。

1988

3月号

特集　コンピュータサウンド"楽"入門
X1/turbo MIDIインタフェイスの製作
MZ-2500 Super Keyboard/VIPサウンドデータ公開
Oh!X LIVE SPECIAL 組曲「Ys」/Raspberry Dream　他
THE SOFTOUCH Might and Magic/Hyper UD
オブジェクト指向のゲームプログラミング
X68000BASIC入門 奇襲アニメ作戦
X68000あなたの知らない世界 未公開IOCSの解析
全機種共通システム 構造型コンパイラ言語SLANG

4月号

特集　不思議の国のゲーム学
決定！　1987年度GAME OF THE YEAR
ピコピコゲーム春場所/GAME REVIEW 10本　他
新製品 X68000ACE-HD/カラースキャナCZ-8NS I
X68000あなたの知らない世界 microEMACSの移植
●MZ-700　SPACE BLUSTER FX
● LIVE in '88 Moonlight Serenade/Long Night 他
全機種共通システム デバッギングツールTRADE
シミュレーションウォーゲームWALRUS

5月号

特集　BASIC入門「再検証」
BASICの歴史と意義/栄光のHuBASIC
黄金のBASIC入門プログラム/プログラミング用語集
ミュージックプログラマへの道/レイトレーシング
特別企画 言わせてくれなくちゃだワ
●新製品　X68000ACE/ACE-HD
● LIVE in '88 GET WILD/BOOM BOOM/SDI
● SHORT ACCESS 3Dボクシング/マシン語データ文生成
全機種共通システム シューティングゲームELFES

6月号　創刊6周年記念

特集　システム環境を考える
8ビットパソコンの開発環境/Human68kのシステム環境/システムを読むためのアセンブラ入門
特別企画 究極の8ビットパソコン 8RON計画
THE SOFTOUCH X68000用日本語ワープロEW 他
●付録「あぶない福袋」
マシン語体操1·2·3 番外編 Lisp80入門
X68000BASIC入門 捨て身のミュージック
全機種共通システム 構造化言語SLANG入門 他

7月号

特集　実践C言語からの誘惑
入門C言語/実録Cプログラミング/XBAS to C
THE SOFTOUCH ソーサリアン/ゼリアード/アルギースの翼/SUPER大戦略/3大麻雀ソフト 他
●Oh!X LIVE in '88/SHORT ACCESS
新連載 C調言語講座PRO-68K まずはprintfより始めよ
あなたの知らない世界 OS-9/X68000/Sampling PRO-68K
全機種共通システム 構造化言語SLANG 入門(2)
マルチウィンドウドライバMW-I

8月号

特集1 真夏の夜の数値演算
コンピュータの数値表現/応用グラフィック歪められた光/AD PCM音の数学/数値演算プロセッサ用ドライバ 他
特集2 MIDIサウンドプログラミング
MIDIの基礎とボードの製作/MIDI対応シーケンサ
THE SOFTOUCH 新連載　われら電脳遊戯民　他
猫とコンピュータ第26回 ボクはかぐや姫？
新連載　Z80マシン語ゲーム工房
全機種共通システムマルチウィンドウエディタWINER

9月号

特集　半期に一度のグラフィックバザール
CGアニメの手法入門/ワイヤフレームによる3D/X68000スプライト/画像処理の基礎知識/turbo RAY TRACER/MZ-2500用グラフィックエディタDMACS
THE SOFTOUCH C-TRACE68/SAMPLING PRO-68K　他
C調言語講座PRO-68K(3) 謎の低次元グラフィック
MIDI活用テクニック(2) 割り込みによるMIDI通信
Z80マシン語ゲーム工房(2) 応用への基礎固め
全機種共通システム ラインエディタTED-750/WINERの拡張

10月号

特集　百花繚乱ゲームバトルロイヤル
最新ゲーム総登場　ハイドライド3/A列車で行こうII/たんば/熱血高校ドッジボール部/フルスロットル　他
MZ-700用SPACE HARRIER
● Oh!X LIVE 1974(16光年の訪問者)/瑠璃色の地球/二人のゼネレーション/バッハのアリア
MIDI活用テクニック(3)複数の音源を操るテクニック
C調言語講座PRO-68K(4)/Z80マシン語ゲーム工房(3)
全機種共通システム SLANG用拡張ライブラリ/MANKAI

11月号

特集　いまどきのプリンタ活用術
メカニズムを理解しよう/制御コード/文字と図形の混在印字/拡大文字のスムージング/外字登録ツール/S-H COPY/グラフィックのモノクロ出力/X68000のCOPYキー/オリジナル印刷キット/試用レポート
THE SOFTOUCH NEW Print Shop PRO-68K 他
OS-9/X68000入門(1) OS-9ってなに？
● STAR TREK for X68000
全機種共通システム シューティングゲームELFES IV

12月号

特集　パソコンはいま音楽の領域へ
なぜ自動作曲か/心地よい雑音の話/和音の読み方/美しい響きの要素/4分音符は歌い始める/古くて新しい音楽形式/FM音源の仕組み/Melody Box/MusicBASIC
●さよなら Live in '88　バッハ イタリア組曲他6本
●Oh!X　1周年記念特別企画「ちょっとあぶない福袋」
OS-9/X68000入門(2) OS-9のオペレーション環境
Z80マシン語ゲーム工房/C調言語講座PRO-68K
全機種共通システム ソースジェネレータSOURCERY

1989

1月号

特集　いきなり初春からハードウェア
デジタル回路入門/電子サイコロ/乱数発生器/X1turboバンクメモリ拡張/X68000用CP/M-80システム　他
1988年度GAME OF THE YEAR ノミネート作品発表
● MZ-2500用　Hyper Game Book
● LIVE in'89　エンデューロレーサー/アルルの女
●ようこそ，セガ・メガドライブ！！
C調言語講座PRO-68K/Z80マシン語ゲーム工房
全機種共通システム パズルゲーム LAST ONE/FLICK

2月号

特集　マシン語"でじたるざんまい"
アーキテクチャからのマシン語入門/アセンブラへの招待/超入門Z80マシン語活用術/X68000料理教室
THE SOFTOUCH 彩CRONE/Final Ver.3.2　他
● X1/X1turbo用RPG FLAME
Z80マシン語ゲーム工房　最終回 爆発，そして完成へ
C調言語講座PRO-68K　第8回 とおりゃんせなのである
OS-9/X68000入門(3) ついに発売！OS-9/X68000
全機種共通システム 高速エディタアセンブラREDA

日本ソフトバンクの
書籍特約書店

下記の書店の一覧は、日本ソフトバンク書籍特約店として右にある商品の他、新刊もとりそろえております。ご希望の商品がある場合は、下記のお近くの書店にてお買い求め下さい。

(注)現品が売れて補充中の場合もございますので、ご注意下さい。

SOFT BANK

日本ソフトバンク出版事業部

〒102 東京都千代田区九段南2-3-26 ☎03(261)4095

地域	書店名	電話
〈北海道〉		
札幌市	紀伊國屋書店札幌店	011-231-2131
〃	旭屋書店札幌店	011-241-3007
〃	丸善札幌支店	011-241-7252
〃	リーブルなにわ	011-221-3800
〃	富貴堂札幌パルコ店	011-214-2303
〃	ダイヤ書房本店	011-712-2541
〃	ダイヤ書房西店	011-665-6223
旭川市	旭川富貴堂	0166-26-3481
〃	ブックス平和マルカツ店	0166-23-6211
苫小牧市	旭屋書店苫小牧店	0144-36-5185
〈東　北〉		
青森市	成田本店	0177-23-2431
〃	岡田書店	0177-23-1381
弘前市	紀伊國屋書店弘前店	0172-36-4511
〃	ブックイン城東	0172-28-2882
八戸市	伊吉書院	0178-44-1917
盛岡市	東山堂書店本店	0196-53-6464
〃	さわや書店	0196-53-4411
〃	第一書店	0196-53-3355
仙台市	金港堂	022-225-6521
〃	金港堂ブックセンター	022-223-0979
〃	アイエ書店駅前店	022-264-0718
〃	丸善仙台支店	022-266-1127
〃	髙山書店	022-263-1511
〃	ブックスみやぎ	022-267-4422
秋田市	三浦書店	0188-33-8131
山形市	八文字屋	0236-22-2150
福島市	岩瀬書店コルニエツタヤ店	0245-21-2101
〃	博向堂	0245-21-1161
郡山市	東北書店	0249-32-0379
いわき市	ヤマニ書房本店	0246-23-3481
〃	鹿島ブックセンター	0246-28-2222
原町市	文芸堂	0244-22-1720
〈関　東〉		
水戸市	川又書店駅前店	0292-31-0102
〃	ツルヤブックセンター	0292-25-2711
勝田市	武石書店	0292-73-1212
東海村	大野書店	0292-82-2098
鹿島郡	なみき書店	0299-96-1855
土浦市	共栄堂	0298-21-6134
筑波市	丸善筑波大学会館店	0298-51-6000
〃	友朋堂吾妻本店	0298-52-3665
宇都宮市	落合書店オリオン店	0286-34-3777
〃	落合書店東武ブックセンター	0286-34-8271
〃	新星堂宇都宮店	0286-33-2337
小山市	進駸堂駅ビル店	0285-25-1522
前橋市	煥乎堂	0272-23-1211
〃	リブロ前橋店	0272-34-1011
〃	戸田書店前橋店	0272-61-5063
高崎市	学陽書房	0273-23-4055
〃	サカ井書店	0273-62-1500
〃	新星堂高崎店	0273-27-3961
〃	戸田書店高崎店	0273-63-5110
太田市	ナカムラヤ	0276-22-2001
〈首都圏〉		
浦和市	須原屋本店	0488-22-5321
浦和市	須原屋コルソ店	0488-24-5321
大宮市	押田謙文堂	0486-41-3141
〃	ブックセンター押田	0486-47-3141
〃	三省堂ブックポート	0486-46-2600
蕨市	須原屋蕨店	0484-44-1211
川口市	岩渕書店川口店	0482-52-2190
川越市	黒田書店川越店	0492-25-3138
所沢市	芳林堂所沢店	0429-25-5355
〃	いけだ書店所沢店	0429-28-3271
上福岡市	黒田書店上福岡店	0492-66-0120
朝霞市	文教堂朝霞店	0484-76-0107
志木市	新星堂志木店	0484-74-0182
春日部市	文教堂春日部店	0487-52-7666
比企郡	錦電サービス	0492-96-2962
千葉市	多田屋セントラルプラザ店	0472-24-1333
〃	キディランド千葉店	0472-25-2011
習志野市	巌翠堂	0474-72-5011
船橋市	ときわ書房本店	0474-24-0750
〃	リブロ船橋店	0474-25-0111
〃	旭屋書店船橋店	0474-24-7331
〃	芳林堂津田沼店	0474-78-3737
〃	第二巌翠堂	0474-65-0926
柏市	西口アサノ	0471-44-2111
〃	新星堂柏店	0471-64-8551
松戸市	堀江良文堂	0473-65-5121
〃	辰正堂駅ビル店	0473-64-7997
横浜市	有隣堂トーヨー店	045-311-6265
〃	有隣堂東口ルミネ店	045-453-0811
〃	栄松堂相鉄ジョイナス店	045-321-6831
〃	そごうブックセンター	045-465-2111
〃	丸善ブックメイツポルタ店	045-453-6811
〃	有隣堂伊勢佐木店	045-261-1231
〃	有隣堂戸塚店	045-881-2661
〃	文華堂戸塚店	045-864-5151
〃	アーバン文華堂	045-821-5151
〃	文教堂青葉台南口店	045-983-5150
川崎市	有隣堂アゼリア店	044-245-1231
〃	有隣堂川崎BE店	044-200-6831
〃	文学堂本店	044-244-1251
〃	ブックセンター文教堂	044-811-5557
〃	文教堂溝ノ口店	044-811-8258
鎌倉市	島森書店大船店	0467-46-3841
〃	鎌倉書店	0467-46-2619
横須賀市	平坂書房WALK店	0468-25-5537
藤沢市	有隣堂藤沢店	0466-26-1411
〃	リブロ藤沢店	0466-27-0111
〃	文教堂六会店	0466-82-9610
茅ヶ崎市	川上書店ルミネ店	0467-87-3827
平塚市	サクラ書店駅ビル店	0463-23-2751
〃	文教堂四之宮店	0463-54-2880
小田原市	八小堂書店	0465-22-7111
〃	伊勢治書店	0465-22-1366
〃	文教堂小田原店	0465-36-3677
厚木市	有隣堂厚木店	0462-23-4111
大和市	文教堂中央林間店	0462-75-4165
相模原市	文教堂相模大野店	0427-49-0650
〃	文教堂橋本店	0427-74-5581
相模原市	文教堂星ヶ丘店	0427-58-6121
津久井郡	文教堂城山店	0427-82-9278
〈東　京〉		
千代田区	三省堂書店神田本店	03-233-3312
〃	書泉グランデ	03-295-0011
〃	東京堂書店	03-291-5181
〃	旭屋書店水道橋店	03-294-3781
〃	丸善お茶の水店	03-295-5581
〃	巌翠堂	03-291-1362
〃	いずみ神田南口店	03-254-8521
〃	明正堂秋葉原店	03-257-0758
中央区	八重洲ブックセンター	03-281-1811
〃	日本橋丸善	03-272-7211
〃	旭屋書店銀座店	03-573-4936
港区	書原新橋店	03-591-8738
〃	雄峰堂ＮＳ店	03-503-6586
〃	虎ノ門書房本店	03-502-3461
〃	虎ノ門書房田町店	03-454-2571
品川区	芳林堂大井町店	03-474-4946
〃	明屋書店五反田店	03-492-3881
渋谷区	紀伊國屋書店渋谷店	03-463-3241
〃	旭屋書店渋谷店	03-476-3971
〃	三省堂書店渋谷店	03-407-4545
〃	大盛堂書店	03-463-0511
〃	紀伊國屋書店笹塚店	03-485-0131
新宿区	紀伊國屋書店本店	03-354-0131
〃	三省堂書店新宿西口店	03-343-4871
〃	福家書店センタービル店	03-345-1246
〃	福家書店野村ビル店	03-342-0298
〃	新星堂ＮＳビル店	03-344-2055
〃	西武新宿ブックセンター	03-208-0380
〃	芳林堂高田馬場店	03-208-0241
〃	未来堂	03-200-9185
豊島区	旭屋書店池袋店	03-986-0311
〃	芳林堂池袋店	03-984-1101
〃	リブロ池袋店	03-981-0111
〃	三省堂書店池袋店	03-987-0511
〃	新栄堂本店	03-984-2345
〃	新栄堂アルパ店	03-988-0181
台東区	明正堂中通り店	03-831-0191
墨田区	リブロ錦糸町店	03-846-0111
〃	ブックストア・談	03-635-1841
江戸川区	文教堂西葛西店	03-689-3621
大田区	アクトブックスサンカマタ店	03-735-1551
〃	竜文堂大森駅ビル店	03-775-3851
中野区	明屋書店東京本社	03-387-8451
杉並区	ブックセンター荻窪	03-393-5571
〃	書原杉並店	03-313-4778
武蔵野市	紀伊國屋書店吉祥寺東急店	0422-21-5543
〃	弘栄堂吉祥寺店	0422-22-1031
〃	パルコブックセンター吉祥寺	0422-21-8122
調布市	真光書店	0424-87-2222
府中市	啓文堂	0423-66-3151
三鷹市	三省堂書店三鷹店	0422-48-4510
〃	東西書房	0422-46-0275
小金井市	文教堂小金井店	0423-86-0161
国分寺市	三成堂国分寺店	0423-25-3211

展示図書一覧

- MS-DOSいたれりつくせり本 ●1800円
- プレイMS-DOS ●1900円
- UNIX System V プログラマ・ガイド ●12000円
- UNIX System V ユーザ・ガイド ●9800円
- UNIXオペレーティングガイド ●3000円
- C言語の活用理解 ●2000円
- C言語の基礎知識 ●2500円
- C言語の応用50例 ●2300円
- 上級・C言語の応用例50例 ●2400円
- Cプリプロセッサ・パワー ●2200円
- Play the C 上巻 ●1500円
- Play the C 下巻 ●1500円
- Turbo C入門 ●2600円
- 8086アセンブリ言語 ●2800円
- 8086マクロプログラミング ●2600円
- ビギニングMUMPS ●2600円
- マシン語マジックブックII ●2500円
- マシン語プログラミングテクニック ●2000円
- BASICによるプログラミングスタイルブック ●1800円
- ソーティング・ノート ●1900円
- BASICプログラムジェネレータ集 ●2800円
- 98/88スモールビジネスプログラム集 ●2500円
- 88デスクアクセサリ集 ●2000円
- IDOS活用ハンドブック ●2700円
- DISK CHARGE追補版 ●1800円
- フロッピーディスクフル活用ガイド ●2300円
- PC工作入門 ●1800円
- 続・PC工作入門 ●1800円
- PC-286Lブック ●1700円
- 試験に出るX1 ●2800円
- X1テクニカルマスター ●2500円
- X1システム研究室 ●2500円
- 新松ガイド ●2000円
- 一太郎Ver.3ガイド ●2500円
- 新一太郎ガイド ●2300円
- 桐Ver.2ガイド ●2500円
- 花子応用ガイド ●2500円
- Lotus 1-2-3ガイド ●2400円
- P1ガイド ●2300円
- Ninja 2 ガイド ●2300円
- Multiplan Ver.3.1ガイド ●2400円
- ビジュアルラーニングRDB ●2500円
- アセンブラCASL入門 ●2000円
- ハードウェア徹底マスター ●2500円
- FORTRAN徹底マスター ●2800円
- 情報処理の基礎知識 ●1600円
- COBOL徹底マスター ●2900円
- 受験用語ハンドブック ●1800円
- ワープロ文書F・O・P ●1200円
- バイト&ワードの風にのって ●1800円
- ワープロ考現学 ●1200円
- 電子ゲームの「快楽」 ●1200円
- ムーグ・ノイマン・バッハ ●1300円
- RPG幻想事典 ●1500円
- 新明解ナム語事典 ●5000円

国立市	東西書店	0425-75-5061
小平市	文教堂小平店	0423-43-9229
東村山市	文教堂東村山店	0423-96-1115
立川市	オリオン書房ウイル店	0425-27-2311
八王子市	くまざわ書店本店	0426-25-1201
町田市	有隣堂町田店	0427-23-3018
〃	久美堂本店	0427-25-1330
〃	久美堂小田急店	0427-27-1111
〃	久美堂東急ハンズ店	0427-28-2772
〃	文教堂鶴川店	0427-35-4117
〃	文教堂小川店	0427-96-1781
多摩市	くまざわ書店桜ヶ丘店	0423-37-2531
福生市	文教堂福生店	0425-53-7708
〈甲信越・北陸〉		
甲府市	文教堂甲府店	0552-22-4600
長野市	平安堂長野店	0262-26-4545
〃	長谷川書店	0262-26-2122
上田市	平安堂上田店	0268-22-4545
松本市	ブックスロクサン	0263-35-5555
〃	改造社松本駅ビル店	0263-36-3777
飯田市	平安堂飯田店	0265-24-4545
岡谷市	笠原書店	0266-23-5070
諏訪郡	平安堂下諏訪店	0266-28-1111
新潟市	紀伊國屋書店新潟店	025-241-5281
〃	萬松堂	025-229-2221
〃	北光社	025-228-2321
長岡市	覚張書店	0258-32-1139
〃	ブックセンター長岡	0258-36-1360
〃	長岡技大長峰文化	0258-46-6437
山北町	BOOKメディア	0254-77-3850
富山市	瀬川書店	0764-24-4566
〃	清明堂	0764-24-4166
〃	BOOKSなかだ豊田店	0764-32-1353
〃	文苑堂本郷店	0764-22-0552
〃	文苑堂赤江店	0764-33-0321
高岡市	文苑堂	0766-21-0333
〃	文苑堂横田店	0766-21-0431
金沢市	うつのみや片町店	0762-21-6136
〃	書林香林坊本店	0762-20-5011
野々市町	王様の本本店	0762-46-5325
福井市	勝木書店	0776-24-0428
〃	品川書店新田塚店	0776-24-1112
〈東　海〉		
静岡市	静岡谷島屋呉服町本店	0542-54-1301
〃	江崎書店	0542-54-4481
〃	吉見書店	0542-52-0157
〃	戸田書店SBS店	0542-81-5733
〃	戸田書店曲金店	0542-81-5899
沼津市	吉野屋	0559-23-5676
〃	マルサン書店宝塚店	0559-63-0350
富士市	戸田書店富士店	0545-51-5121
清水市	戸田書店本店	0543-65-2345
浜松市	浜松谷島屋連尺本店	0534-53-9121
名古屋市	三省堂書店名古屋店	052-562-0077
〃	星野書店近鉄ビル店	052-581-4796
〃	丸善ブックメイツセントラルパーク	052-971-1231
〃	日進堂上前津店	052-263-0550
名古屋市	三洋堂パソコンショップΣ	052-251-8334
〃	三洋堂いりなか本店	052-832-8202
〃	ちくさ正文館本店	052-741-1137
〃	白樺書房西店	052-774-7223
豊橋市	精文館	0532-54-2345
岡崎市	ブックス鎌倉	0564-54-1822
豊田市	三洋堂梅坪店	0565-35-2334
刈谷市	三洋堂刈谷店	0566-24-1134
春日井市	三洋堂勝川店	0568-32-7806
岐阜市	自由書房	0582-65-4301
大垣市	大洞堂ブックス258	0584-81-2553
〃	大洞堂岐大バイパス店	0584-74-7766
一宮市	三洋堂一宮店	0586-77-5734
可児市	三洋堂可児店	0574-63-2334
多治見市	三洋堂多治見店	0572-24-0340
津市	別所書店11ビル店	0592-24-1014
四日市市	文化センター白揚	0593-51-0711
鈴鹿市	シェトワ白揚スズカ	0593-82-5221
〈近　畿〉		
京都市	駸々堂京宝店	075-223-1003
〃	アバンティ・ブックセンター	075-682-5031
〃	オーム社書店河原町店	075-221-0280
〃	ジュンク堂京都店	075-252-0101
奈良市	駸々堂大丸店	0742-26-6241
大阪市	旭屋書店本店	06-313-1191
〃	紀伊國屋書店梅田店	06-372-5821
〃	オーム社書店大阪店	06-345-0641
〃	駸々堂京橋店	06-353-3209
〃	駸々堂心斎橋店	06-251-0881
〃	旭屋書店ナンバ店	06-644-2551
〃	ナンバブックセンター	06-644-5501
〃	旭屋書店アベノ店	06-631-6051
〃	ユーゴー書店	06-623-2341
〃	河村書店	06-951-2968
枚方市	水嶋書房京阪デパート店	0720-51-3432
高槻市	コーベブックス西武高槻店	0726-83-1766
東大阪市	ヒバリヤ書店本社	06-722-1121
神戸市	ジュンク堂センター街店	078-392-1001
〃	ジュンク堂サンパル店	078-252-0777
〃	海文堂書店	078-331-6501
〃	日東館書林	078-391-8701
姫路市	新興書房	0792-85-3344
〃	誠心堂書店	0792-81-2055
和歌山市	宮井平安堂	0734-31-1331
〃	帯伊書店	0734-22-0441
〈中　国〉		
岡山市	紀伊國屋書店岡山店	0862-32-3411
〃	丸善岡山支店	0862-31-2261
津山市	津山ブックセンター	08682-6-4047
広島市	紀伊國屋書店広島店	082-225-3232
〃	丸善広島支店	082-247-2251
〃	金正堂	082-248-3715
〃	積善館	082-248-3151
尾道市	啓文社尾道店	0848-37-5151
福山市	啓文社福山店	0849-22-3111
〃	ブックシティ啓文社	0849-25-0050
〃	啓文社コア	0849-41-0909
山口市	五十部誠文堂	0839-24-6630
〃	文栄堂	0839-22-5611
下関市	中野書店	0832-22-6181
宇部市	京屋書店	0836-31-2323
〃	末広書店	0836-31-0086
防府市	誠文堂国衙店	0835-25-1988
光市	三文字屋	0833-71-0251
鳥取市	富士書店	0857-23-7271
松江市	園山書店	0852-21-4167
〈四　国〉		
徳島市	小山助学館本店	0886-54-2135
〃	小山助学館東口店	0886-25-1380
〃	森住丸善	0886-23-3228
高松市	宮脇書店本店	0878-51-3733
丸亀市	宮脇書店丸亀店	0877-22-5533
松山市	紀伊國屋書店松山店	0899-32-0005
〃	明屋書店本店	0899-41-4141
〃	明屋書店大街道店	0899-41-4242
〃	丸三書店	0899-31-8501
新居浜市	明屋星原店	0897-44-4000
宇和島市	明屋宇和島店	0895-23-1118
高知市	金高堂	0888-22-0161
〈九州・沖縄〉		
福岡市	紀伊國屋書店福岡店	092-721-7755
〃	りーぶる天神	092-713-1001
〃	積文館新天町店	092-781-2991
〃	福岡金文堂朝日ビル店	092-431-1094
〃	福岡金文堂デイトス店	092-451-6175
〃	福岡金文堂アニマート原	092-844-0088
北九州市	ナガリ書店	093-521-1044
〃	金栄堂	093-531-3685
〃	旭屋書店北九州店	093-631-6421
〃	井筒屋ブックセンター	093-641-0131
〃	カルパーク平野	093-661-7988
〃	白石書店本城店	093-601-2200
久留米市	エマックスたがみ	0942-33-1841
飯塚市	BOOKリード	0948-25-7266
大分市	パルコブックセンター大分店	0975-35-0643
〃	本町晃星堂	0975-33-0231
別府市	明林堂	0977-23-2183
宮崎市	中央、田中書店	0985-24-3511
〃	寿屋宮崎店	0985-27-4111
佐賀市	金華堂北バイパス店	0952-32-1965
〃	積文館デイトス店	0952-23-7155
長崎市	メトロ書店	0958-21-5453
〃	好文堂	0958-23-7171
佐世保市	金明堂書店	0956-22-4214
熊本市	紀伊國屋書店熊本店	0963-22-5531
〃	長崎書店	0963-53-0555
人吉市	明屋人吉店	0966-22-5486
鹿児島市	春苑堂ブックプラザ	0992-25-3200
〃	ブックスみすみ	0992-57-1011
那覇市	球陽堂書房ビル店	0988-63-3752
〃	文教図書	0988-62-1201

M·A·G·A·Z·I·N·E·S 日本ソフトバンク

月刊
Oh!PC
3月号
500円
好評発売中!

特集 パーソナル・パブリッシング&プレゼンテーション
DTPソフトの実力を探る
第2特集 続・初めてMS-DOSを触るあなたへ
初心者向けMS-DOS入門
●元気一杯! VA ●ツール&ユーティリティWho's Who
●C言語プログラミング ●Soft WATCHING
■ハンディスキャナ活用術 ■D-Shellフィールドノート
■IDOS-GADGETRYスペシャル

月刊
Oh!FM
3月号
540円
好評発売中!

他誌に先駆けて贈る
特集 富士通32ビットニューマシンのすべて!!
●ミュージックツール FMオルゴール
●シューティングゲーム Zearan
●OS-9用ファイルセレクタfs
●MS-DOSプリンタユーティリティ
■6809マシン語道場 ■BASICプログラム工房
■Let's PLAY Computer MUSIC!! ■谷山浩子のエッセイ

月刊・コンピュータ技術者必携 第2種・第1種・特種受験
情報処理試験
好評発売中!
3月号
特別定価680円

特集1 2種合格の基礎固め 計算問題に強くなる!
数の表記法と基数変換/磁気ディスク装置の計算/信頼性と稼動率他
特集2 1種必修テーマ集中講座 データ通信の基礎知識
データ伝送方式/データ通信回線/伝送制御手順/データ通信サービス
△カラー受験ゼミ 最新入出力装置
△続・コンピュータ最前線 PC-98の牙城に挑む「5530Z」
[昭和63年度10月情報処理技術者試験] 1種全合格者
■特別付録 2種・1種午後問題対策

月刊
Beep
MAGAZINE FOR GAME KIDS

3月号
420円
好評発売中!

特集1 「ゲーム界」サクセスストーリー
ゲーム界でサクセスするためには!?
偉大な先達の足跡をたどってみよう!
特集2 ヒットゲームのトレンド大研究!
いまどんなゲームが売れているか!
●徹底マスター ファミコン/PCエンジン/メガドライブ/マークIII
●サウンドマーケット ●今月のおすすめガイド
●「ドラゴンクエストIV」最新情報

月刊
THE COMPUTER
コンピュータ時代を読むトレンド・マガジン
3月号
580円
好評発売中!

特集 CPUパワーがパソコンを変える
●KEYMAN U.S.A.パソコンを次世代へつなぐネットワークの雄 ノベル社長レイ・ノーダ
●THE TEST 低価格プリンタのBEST CHOICE
●田原総一朗のコンピュータ・ルポ インテルジャパン社長W.O.ハウ
●ヒット商品開発ストーリー ローランド「ミュージくん」
シリーズ ■コンピュータ時評 ■パソコン辛口コラム
■COMジャーナル ■ハイテク政治経済学
特別付録 システム手帳用リフィル 目的別MS-DOSカード

SHARP

シャープ パソコンフォーラム'89 in 赤坂

3月25日(土) 13:00〜18:00 ／26日(日) 10:00〜17:00

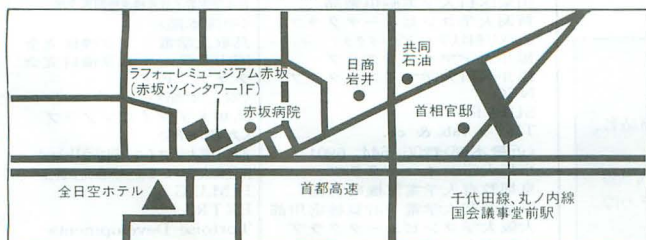
ラフォーレミュージアム赤坂

（赤坂ツインタワービル1F）

入場無料

パソコンのすべてと最先端テクノロジー展

先着5,000名様にご来場記念品進呈

X68000もAXラップトップも… 新製品を含めたパソコンのすべてと、そのシステム展開、またカラー液晶技術など最先端テクノロジーも盛りだくさんにご紹介。ソフトハウス20数社のご出展もいただいて、最新アプリケーションもご体験いただけます。ぜひご来場ください。

パソコンコメンテイター
高橋雄一

ドクターパソコン
宮永好道

パソコンエンターテイナー
山下　章

●各先生をお招きしての各種イベントも用意しています。

主催：シャープエレクトロニクス販売㈱／東京中央シャープ販売㈱／後援：シャープ㈱

問合わせ先　〒162　東京都新宿区市ヶ谷八幡町8番地　シャープエレクトロニクス販売㈱　首都圏統轄営業部　TEL 03-266-8248㈹

究極美表現

エキサイト Xシリーズ第1弾！

X68000初の完全オリジナル・シューティングゲーム

D-RETURN

史上空前！
230名
モニター結果による
全面改良実施！

近日発売!!

対応機種：X68000（5インチ2HD）2枚組

¥5,980

開発者：神戸大学情報統計部 前部長　赤坂 賢洋(NOP)

斜めスクロール！逆スクロール！
5重スクロール！半透明！
X68000の性能をフルに生かした
究極のシューティングゲーム！
全8面、各面ごとにBGMが違う
オリジナルBGM40曲以上使用！
各面ごとにボスキャラ登場！
コンフィグレーション画面にて、
スピード、難易度、自機数、ミサイルの種類、開始面などの設定が自由！

●他のゲームとは、ここが違う！

戦争とは、一体何なのかを問いかける感動のストーリー展開。
コンフィグレーション画面の設定により、好きな難易度で、好きな面からのゲームプレイが可能。
しかし、全8面のボスキャラを攻略し、涙と感動のエンディングを体験できるのは、ごく一握りの技を極めたゲーマーのみ。
敵キャラから発射される爆弾が異常に多いので、繰り返しプレーしているうちにシューティング力が確実にアップ。

◆ご注意…バックアップされたソフトでは、エンディングに達することはできません。

AN ADVENTURE GAME INTERPRETER

Cyber Writer

電脳作家 Ver 2.0

対応機種：X68000(5インチ2HD)2枚組 ¥5,980

開発者：神戸大学情報統計部　部長　村尾 元

電脳作家は、専用の言語で書かれたシナリオをX68000上で、コマンド選択式アドベンチャーゲームの形で実行する一種のインタプリタです。Ver2.0では、OPMによる音楽演奏やPCMによる音声出力も可能となり、より良質のアドベンチャーゲームが作れるようになりました。便利なグラフィックツールに加え、買ったその日から遊べるサンプルシナリオ付きです。

電脳作家グラフィック&ミュージックライブラリー集 ¥3,980

対応機種：X68000(5インチ2HD)2枚組
制作者：神戸大学情報統計部　細見格・赤坂賢洋

◆グラフィックファイル10ファイル、ミュージックファイル39ファイル収録。

第3回アドベンチャーゲーム・シナリオコンテスト実施中！

電脳作家で動作する自作アドベンチャーゲームシナリオを募集します。
締切り　1989年4月30日（消印有効）

シナリオコンテスト入選作品通信販売開始

EVIL EYE 作：三上潤一郎（Jun. M. Win）　Ver2.0対応
価格1000円（日コン連企画㈱までお申し込みください。）

郵送品貼付切手には、オール記念切手使用！

日コン連SOFT通信販売のご案内

現金書留または、郵便振替（大阪5-4873 日コン連企画株式会社）で、希望商品名、対応機種名、数量明記の上、お申し込みください。（送料はサービス）
なお、現金書留でお申し込みの場合、20円分余分に入れ、端数をなくす（例5,980円→6,000円）と、重量が軽くなり、送料が安く（520円→410円）なります。その際のおつりは、商品発送時に同額の記念切手でお返し致します。

お知らせ！

大好評！日コン連パナコム受験SIG開催中

期間 3月31日まで、上新電機J&P HOTLINE上で
参加大学：日コン連加盟の全国公立大学29大学と東京大・三重大・島根大計32大学
J&P HOTLINE ID収得希望者は、日コン連企画(株)まで3,500円を現金書留、郵便振替、小為替でお送りください。折り返し、スタータキットをお送りいたします。

日本コンピュータクラブ連盟 加盟団体募集！
加盟費・会費等一切不要。

●加盟団体一覧（1989.1月現在）

〈北海道・東北支部〉
S.M.S.C
Some times
MAC
〈関東本部〉☎045-784-9441
東京水産大学コンピュータクラブ
東京学芸大学教育工学研究会
横浜市立大学パソコンクラブ
NEW MZM
パーソナルコンピュータクラブPeke
CSS
〈中部本部〉
名古屋大学コンピュータ研究会
名古屋工業大学コンピュータ倶楽部
名古屋市立大学システム研究会
愛知教育大学SF研究会ポッツ
豊橋技術科学大学コンピュータクラブ
岐阜大学コンピュータクラブ
浜松医科大学マイコン研究会
山梨大学電子計算機研究会
山梨医科大学電脳倶楽部
新潟大学コンピュータクラブ
富山医科薬科大学コンピュータクラブ
福井大学マイコンクラブ
福井医科大学マイコンクラブ
NEO
SUPER
TRY-x lab. & co.
〈近畿本部〉☎06-644 6901
京都大学マイコンクラブ
京都教育大学電算機研究部
京都産業大学電子計算機応用部
大阪大学コンピュータクラブ
大阪市立大学マイコン研究会
関西大学情報処理技術研究会
大阪電気通信大学電子計算組織研究会
近畿大学電気技術部
神戸大学情報統計部
県立神戸商科大学電子計算機研究会
神戸女学院大学マイコン研究会
甲南女子大学マイコン研究同好会
滋賀大学電子計算機クラブ
和歌山大学マイコン研究会
BLACK—BOX
UNLINK
日本コンピュータチェス協会
Traveling Club
T.A.C
J.K.M.C
蝶
NERKEY
Do—GA
REVOLB
RPG CLUB
日本学生電子計算機連盟関西支部
〈中国本部〉
鳥取大学電子計算機研究会
岡山大学電子計算機研究会
クラブN.F.T
〈四国本部〉
高知大学マイコンクラブ
〈九州本部〉
九州工業大学マイコン同好会Hybrid
鹿児島大学コンピュータ研究委員会一来夢
H.M.U.G
EXTRA
Tortoise Developments

日コン連SOFT保証

日コン連SOFTのディスク内容をお客様が破損された場合、そのディスクと300円分の切手を同封してお送り頂ければ、折り返し、新しいディスクをお送りしています。

●問い合わせ先・申し込み先

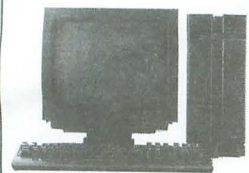
日コン連 SOFT

日コン連企画株式会社・日本コンピュータクラブ連盟(共通)
〒556 大阪市浪速区難波中2-4-3 村上ビル
TEL　06(644)6901 ㈹

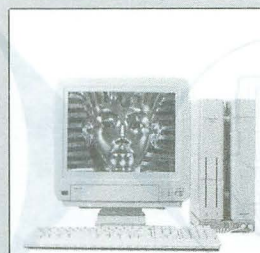
信用と実績を誇る BASIC HOUSE

北関東最大の68000専門店

SHARP SONY NEC 横河ヒューレットパッカード YHP Apple Computer

BASIC HOUSEおすすめ特別セット

Hyper COBURA SET A

CZ-601C	￥319,800
CZ-603D	￥ 84,800
ハンデープリンタ	￥ 24,800
メロディBOX	￥ 16,800
当社オリジナルソフト(7本)	￥ 74,400
5インチ2HD10枚PDSソフトサービス	
定価合計	￥520,600
超特価	￥389,000 〒1,000

長期クレジット　頭金9,000　12,800×36回

Hyper COBURA SET B

CZ-611C	￥399,800
CZ-603D	￥ 84,800
ハンディープリンタ	￥ 24,800
メロディーBOX	￥ 16,800
当社オリジナルソフト(7本)	￥ 74,400
5インチ2HD10枚PDSソフトサービス	
定価合計	￥600,600
超特価	￥459,000 〒1,000

長期クレジット　頭金9,000　15,200×36回

BASIC HOUSEオリジナル

X68000シリーズ

B6-6301 BASIC拡張関数パッケージ	￥ 9,800
B6-6302 CP/M-68K エミュレータ	￥19,800
B6-6303 アイコンエディタ	￥ 4,800
B6-6304 ディスクキャッシャー	￥ 6,800
B6-6305 C言語ライブラリ	￥ 6,800
B6-6306 BASIC拡張関数パッケージC言語ライブラリ付	￥14,800
B6-6307 TOYS & TOOLS	￥ 6,800
HANDY PRINT jack	￥24,800
MELODY BOX MIDIインターフェース	￥16,800
KGB-X68ADC 16ch12ビットA/D変換ボード	￥128,000
KGB-X68PIO アイソレーション16Bit入出力ボード	￥68,000
KGB-X68UNB ユニバーサルボード	￥ 6,800

MZシリーズ

B7-2501 PC-8801→MZ-2500テキストコンバータ	￥ 3,000
B7-2502 PC-8001→ 〃	￥ 3,000
B7-2503 PC-6001→ 〃	￥ 3,000
B7/2504 FM-77 → 〃	￥ 3,000
B7-2505 MSX → 〃	￥ 3,000
B7-2506 S1/L3 → 〃	￥ 3,000
KGB-MZ1 超低価格計測制御ボード	￥15,500

X1/X1turboシリーズ

KGB-X1S 低価格アナログデジタル入出力ボード	￥19,800
KGB-HDIF X1turbo専用ハードディスクインターフェースボード	￥16,000
KGB-PIO 高級絶縁型パラレル入出力ボード	￥42,000

KGB-AD12 高級16ch 12BitA/D変換ボード	￥118,000
KGB-DA4 高級4ch 12BitD/A変換ボード	￥98,000
KGB-488 GP-IBインターフェース(マニュアルソフト付)	￥58,000
B6-3301 PC98↔X1turbo相互ファイルコンバータ	￥ 4,800

PC-8801/PC-9801シリーズ

KGB-PC1 KGB-MZ1のPC-8801版	￥15,500
KGB-98S PC-9801シリーズアナログ入出力	￥19,800
デジタル入出力ボード(D/A付)	￥25,000

Macintoshシリーズ

PRINT jack プリンタドライバー	￥38,000
MOUSE jack マウスドライバー	￥ 4,800
MELODY BOX MIDIインターフェース	￥19,800

X68000 SOFT HARDプライスリスト　BASIC HOUSE特価CALL

ハードウエア

CZ-6BE1 1MB内蔵RAM(CZ-600用)	￥35,000
CZ-6BE1A 1MB内蔵RAM(ACE用)	￥38,000
CZ-8NS1 カラーイメージスキャナー	￥188,000
CZ-6BN1 スキャナーパラレルI/F	￥29,800
CZ-6BC1 FAXボード	￥79,800
CZ-6BP1 数値演算プロセッサーボード	￥79,800
CZ-6BU1 ユニバーサルI/Oボード	￥39,800
CZ-6BG1 GP-IBボード	￥59,800
CZ-8NT1 トラックボール	￥13,800
CZ-6BF1 RS-232Cボード(増設用)	￥49,800
CZ-6EB1 拡張I/Oボックス	￥88,000
KGB-X68 UNB ユニバーサルボード	￥ 6,800
KGB-X68PIO アイソレーションI/Oボード	￥68,000
KGB-X68ADC 16ch12BitA/D変換ボード	￥128,000
KGU-HDPR ハンディープリンタ	￥24,800
KGU-X68MD MIDIインターフェース	￥19,800
CZ-6BM1 MIDIボード	￥26,800

ソフトウエア

CZ-212BS ビジネスプロPRO-68K	￥68,000
CZ-220BS DATA PRO-68K	￥58,000
CZ-226BS CARD PRO-68K	￥29,800
CZ-214MS SOUND PRO-68K	￥15,800
CZ-213MS MUSIC PRO-68K	￥18,800
CZ-215MS Sampling PRO-68K	￥17,800
CZ-221HS NEWprint shop	￥19,800
CZ-223CS Communication PRO-68K	￥19,800
CZ-211LS C Compiler PRO-68K	￥39,800
EW 日本語ワープロ	￥38,000
ビジレスAD68K 表集計データベース	￥98,000
KamiKaze 統合化ソフト	￥68,000
C-TRACE グラフィックツール	￥68,000
Z'S STAFF PRO-68K 〃	￥58,000
Hyper UD 〃	￥16,800
X Link PRO-68K 通信ソフト	￥19,800
CZ-237MS music Studio PRO-68K	￥25,800

ゲーム

ハウメニロボット	￥ 9,500
魔神宮	￥ 7,800
グランドマスター	￥ 9,800
DOME	￥ 9,800
上　海	￥ 6,500
スペースハリア	￥ 6,800
T.D.F	￥ 6,800
源平討魔伝	￥ 7,800
ザ・ラスベガス	￥ 9,800
レリクス	￥ 7,200
マンハッタンレクイエム	￥ 7,800
桃太郎伝説	￥ 7,800
ツインビー	￥ 7,800
アルカノイド	￥ 7,800
沙羅曼蛇	￥ 8,800
フルスロットル	￥ 8,800
熱血高校ドッジボール部	￥ 7,800
A列車で行こうII	￥12,800
ザ・リターンオブイシター	￥ 7,800

限/定/特/価

￥69,000

X68000用MIDIインターフェース
MELODY BOX(X68000)　￥16,800

￥69,000+￥16,800=￥85,800
限定20台特価　￥72,000　送料￥1,000

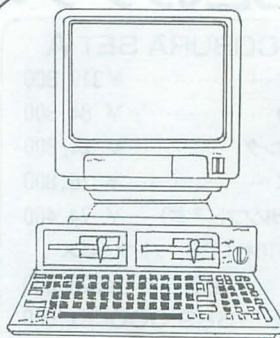
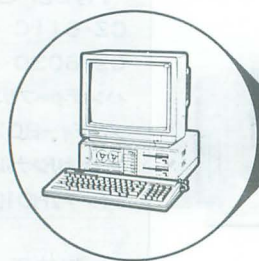
社員募集

職　種：ソフトウエア技術者
　　　　ハードウエア技術者
　　　　セールスエンジニア
　　　　マイコンショップ店長候補
資　格：35歳位までの男女経験者
給　与：能力、経験、年令を考慮し優遇
待　遇：昇給年1回、賞与年2回
　　　　各種保険年金制度、海外研修制度あり
休　日：日、祝祭日、年末年始、夏季

勤務地：本社及び各事業所
応募方法：履歴書をご送付ください。
　　　　後日面接日を通知いたします。
　　　　応募の秘密は厳守します。

株式会社**計測技研**　本社営業部／マイコンショップ／通販部　宇都宮市竹林町503—1　TEL0286-22-9811　FAX0286-25-3970

マイコンショップ BASIC HOUSE　お申し込み・お問い合せは ☎0286-22-9811(代)

超高速の橋出現！

X1・X1turbo用

SUPER DEVICE MONITOR "T"

BLUE SKYはコンピュータ通信にオブジェクトデータの橋を架けました。今迄はRS-232Cでオブジェクトデータを通信する時は，アスキーデータに変換して行っていたコンピュータ通信を，直接オブジェクトデータのままで，しかも，特殊なデータ圧縮を施して，今迄にない超高速で通信する事が出来るX1turbo用の『SUPER DEVICE MONITOR "T"』を開発しました。既に好評発売中のMZ用の『SUPER DEVICE MONITOR "T"』とはRS-232Cにより双方向の超高速通信が出来ます。

エディト機能も呼び出したセクターを豊富なコマンドを使ってワープロ感覚で自在に変更・書き込み等のデータの編集が簡単に出来ます。アクセス出来るディバイスもハード・ディスク，MS-DOSやX68000で使用しているフォーマットの2HDのディスクなど各コンピュータに接続された殆どのディバイスをエディトする事が出来ます。

★任意のディバイスから他のディバイスへセクター単位で高速転送が出来る。

★任意のセクターをほぼ瞬間的に縦・横チェックサムとキャラクターダンプ付き表示が出来る。

★エディット機能はワープロ感覚で表示したセクターのオブジェクト・データを1バイト単位で変更・複写等多彩なエディト機能を備えている。

★turbo内のBIOS用ROMやturboZII標準装備の内部増設メモリーにも直接アクセス出来る。（turboのみ）

★任意のディバイスの複数のセクターを他のディバイスと比較・照合が出来る。

★キャラクターダンプは漢字の表示も出来る。（X1は除く）

★RS-232Cのボーレートの変換はボタン一つで切り替えられる。

★X1フォーマットやMZフォーマットのディスクがアクセス出来る。

★X68000やMS-DOSフォーマットのディスクにもアクセス出来る。（turboのみ）

★255バイト迄のデータを任意のディバイスの複数のセクターから検索する事が出来る。

★キャラクターダンプで表示出来る漢字には区点・JISの表示も出来る。（turboのみ）

★2HD及び2DDのディスクもアクセス出来る。（turboのみ）

★RS-232Cを使って他のコンピュータとの間で相互に特殊なデータ圧縮法に因り複数のセクターのオブジェクト・データを通常の最高32倍（理論値）の超高速での転送が出来る。（X1は除く）

SUPER DEVICE MONITOR "T"
（turbo用の2HDは受注生産）

機種	メディア		価格
X1	5″	2D	10,000円
X1turbo	5″	2D/2HD	13,000円
MZ-2500・2800	3.5″	2DD	13,000円

ロードに長時間かかる多分割のテープ版のゲームがボタン操作一つで何本も1枚のディスクに整理が出来て表示したリストから遊びたいゲームを指定すると一瞬でロード出来る『EXTRA HYPER+α』もあります。

EXTRA HYPER + α X1・X1turbo MZ-2000・2200・2500 3″・5″ 3.5″・5″ 各14,000円

BLUE SKY Co.

▶お求めは全国の有名マイコンショップでどうぞ。
通信販売をご希望の方は当社へ直接、商品名・機種名・メディア名・住所
氏名・電話番号を明記の上、現金書留にてお申し込みください。（送料無料）

株式会社 BLUE SKY
〒411 静岡県三島市加茂16-4
☎0559-72-6710

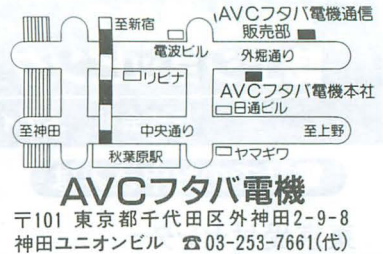
秋葉原価格でローンができます
電気の街秋葉原で
24年
の信用!!

AVCフタバ

AUDIO・VIDEO・COMPUTER

☎03(253)7661

至新宿　電波ビル　リビナ　AVCフタバ電機通信販売部　外堀通り　AVCフタバ電機本社　日通ビル　至神田　中央通り　至上野　秋葉原駅　ヤマギワ

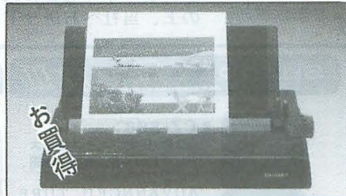
AVCフタバ電機
〒101 東京都千代田区外神田2-9-8
神田ユニオンビル ☎03-253-7661(代)

今すぐ もよりの電話から	仙 台 022-264-3704	名古屋 052-452-3271	広 島 082-295-6873
札 幌 011-611-5104	新 潟 0252-75-4175	大 阪 06-311-3931	福 岡 092-481-2494

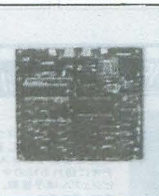
X68000の情報のすべて!(当店はX68000の認定代理店です。お気軽にご相談下さい)

CZ-8PC2 標準価格¥69,800(今、プリンターを探してる人へ フタバのお買得情報。)

C.G.やカラーイメージボードで取り込んだ映像を7色で鮮やかにハードコピー。JIS第1水準・第2水準の漢字をはじめ、多彩な文字種が使え、文書作成には24×24ドットの高品位で対応。
●オプション黒色リボンカセット CZ-8PC-1 ¥700
　　　　　カラーリボンカセット CZ-8PC-2 ¥800
※信号ケーブルおよび黒色/カラーリボンカセット各1個付

●CZ-8PC2の主な仕様

印字方式	ドットマトリクス・ノンインパクト熱転写
文字種類	288種(英数・記号・カナ(またはひらがな)・その他)
文字ドット構成	19(タテ)×15(ヨコ)ドット(パイカ文字、エリート文字)、19(タテ)×10(ヨコ)ドット(縮小文字) 24(タテ)×24(ヨコ)ドット(漢字)、24(タテ)×12(ヨコ)ドット(半角文字)
印字速度	45字/秒(パイカ文字)、30字/秒(漢字)
電源・消費電力	AC100V、50/60Hz、40W(印字動作中)、15W(待機中)
外形寸法・重量	幅365×奥行253×高さ65(mm)、約3.5kg

特価 ¥4?,800
お支払例 ¥ 8,168× 6回 ¥ 5,038×10回
¥ 4,236×12回 ¥ 3,450×15回

CZ-6BC1

FAXボード。拡張I/Oスロットに装着し電話回線を利用してデータ一通信を行う事ができる。

標準価格……¥79,800
特価 ¥6?,000
お支払例 ¥ 5,920×12回 ¥ 4,071×18回
¥ 3,712×20回 ¥ 3,147×24回

CA-40LG(キャラベルデータシステム)

万全の出荷検査によるパーフェクトな品質保証。
平均35msc以下。
この際一気に40M増設を考えては?
CA-40LG
標準価格…¥138,000
特価 ¥103,000
お支払例 ¥ 9,528×12回 ¥ 6,552×18回
¥ 5,064×24回 ¥ 3,548×36回

実装密度をさらに追求、信頼性を高めたハイコストパフォーマンスFDモデル。HDモデル。

X68000 PERSONAL WORKSTATION ACE・ACE HD

FDD搭載タイプの場合

CZ-601C	¥319,800
CZ-611D	¥145,000
合 計	¥464,800

激安

CZ-601C	¥319,800
CZ-601D	¥119,800
合 計	¥439,600

何故か安い

CZ-601C	¥319,800
CZ-603D	¥ 84,800
合 計	¥404,600

(こんな表示で申し訳ない! はっきり書くとおこられます。)

(注)CZ-611Dは0.31mmと、グラフィックをされる方にはすぐれたモニターです。CZ-601DはTVも写る、皆様に人気のあるモニターです。CZ-603DはTVは写りません、パソコンのディスプレイとしてご使用下さい。

※写真はCZ-601C-BK+CZ-603D-BK

HDD搭載タイプの場合

CZ-611C	¥399,800
CZ-611D	¥145,000
合 計	¥544,800

バカに安い

CZ-611C	¥399,800
CZ-601D	¥119,800
合 計	¥519,600

群を抜いて安い

CZ-611C	¥399,800
CZ-603D	¥ 84,800
合 計	¥484,600

(申し訳ないが値段は言えない程安いんです。)
分割の価格はどうぞお気軽にお問合せ下さい。

初めて買われる方へ。

HDD搭載タイプと云うのは20MBのハードディスクが付いています。
とりあえず初心者には必要ないだろうと思われますので601C+601Dをお勧めします。又他のオプションと併売する場合は充分な値引も考慮します。

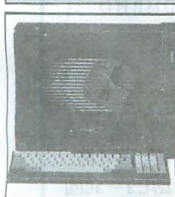
もちろん、分割払いもできます。

■本体+キーボード+マウス・トラックボール CZ-601C-GY(グレー)・-BK(ブラック)標準価格319,800円
■14型カラーディスプレイCZ-603D-GY(グレー)・-BK(ブラック)標準価格84,800円(チルトスタンド同梱)

型番	品名	標準価格	販売価格	お支払例
CU-14BD	ディスプレイ	¥ 64,800	¥ ?7,800	¥ 3,601×15回
CU-14ED	ディスプレイ	¥ 79,800	¥ ?2,600	¥ 3,346×18回
CU-14AD	ディスプレイ	¥ 84,800	¥ ?3,800	¥ 3,422×18回
CU-21CD	ディスプレイ	¥139,800	¥1?0,000	¥ 5,408×24回
CZ-820D	ディスプレイ	¥ 79,800	¥ ?4,800	¥ 3,375×15回
CZ-880D	ディスプレイ	¥109,800	¥ 8?,000	¥ 4,081×24回
CZ-603D	ディスプレイ	¥ 84,800	¥ ?3,800	¥ 3,137×24回
CZ-601D	ディスプレイ	¥119,800	¥ ??,000	¥ 3,203×36回
CZ-611D	ディスプレイ	¥145,000	¥1?3,000	¥ 3,892×36回
BF-68PRO	CRTフィルター	¥ 19,800	¥ 1?,800	現金一括払
CZ-502F	FDD(2DD)	¥ 99,800	¥ 7?,000	¥ 3,172×30回
CZ-503F	FDD(2D)	¥ 49,800	¥ 3?,800	¥ 3,219×12回
CZ-6BE1A	1MB 増設RAMボード	¥ 38,000	¥ 3?,800	¥ 3,388×10回
CZ-6BE2	2MB 増設RAMボード	¥ 79,800	¥ 6?,000	¥ 4,071×18回
CZ-6BE4	4MB 増設RAMボード	¥138,000	¥1?8,000	¥ 3,720×36回

型番	品名	標準価格	販売価格	お支払例
AN-8TU	RGBシステムチューナ	¥ 35,800	¥ 28,800	現金一括払
CZ-8PK7	プリンタ(80桁)	¥122,000	¥ 9?,000	¥ 3,238×36回
CZ-8PK8	プリンタ(136桁)	¥152,000	¥1?7,000	¥ 3,169×48回
CZ-8PK9	プリンタ(80桁)	¥ 89,800	¥ ?0,000	¥ 3,442×24回
CZ-6VT1	カラーイメージユニット	¥ 69,800	¥ 5?,000	¥ 3,562×18回
CZ-8BV2	カラーイメージボード	¥ 39,800	¥ 31,800	¥ 3,498×10回
AN-160SP	アンプ内蔵スピーカー	¥ 59,800	¥ ?8,000	¥ 3,053×18回
CZ-8BS1	FM音源ボード	¥ 23,800	¥ 19,800	現金一括払
CZ-6BN1	スキャナ用パラレルボード	¥ 29,800	¥ 24,000	現金一括払
CZ-6BU1	ユニバーサルIOボード	¥ 39,800	¥ 3?,000	¥ 3,520×10回
CZ-6BG1	GP-IBボード	¥ 59,800	¥ 4?,000	¥ 3,053×18回
CZ-8TM1	モデム	¥ 29,800	¥ 25,000	現金一括払
CZ-8TM2	モデム	¥ 49,800	¥ ?9,000	¥ 3,608×12回
CZ-8NT1	トラックボール	¥ 13,800	¥ 12,500	現金一括払
CZ-6SD1	システムラック	¥ 44,800	¥ 3?,800	¥ 3,312×12回

型番	品名	標準価格	販売価格	お支払例
CZ-6BF1	増設RS232Cボード	¥ 49,800	¥ 4?,000	¥ 3,013×15回
CZ-6BP1	数値プロセッサボード	¥ 79,800	¥ 6?,000	¥ 3,147×24回
CZ-6EB1	I/Oボックス	¥ 88,000	¥ 7?,000	¥ 3,442×24回
CZ-234LS	AI開発ツール	¥188,000	AVC特価	
CZ-219SS	OS-9	¥ 29,800	AVC特価	
CZ-227BS	TOP財務会計	¥200,000	¥1?8,000	¥ 4,279×48回
CZ-213MS	MUSIC PRO-68K	¥ 18,800	¥ 15,800	現金一括払
CZ-214MS	SOUND PRO-68K	¥ 15,800	¥ 13,800	現金一括払
CZ-212BS	ビジネス PRO-68K	¥ 68,000	¥ 5?,000	¥ 3,435×18回
CZ-211LS	Cコンパイラ PRO-68K	¥ 39,800	¥ 3?,000	¥ 3,520×10回
CZ-141SF	NEW-Z BASIC	¥ 18,800	¥ 15,800	現金一括払
CZ-137SF	turboZ's STAFF	¥ 19,800	¥ 16,800	現金一括払
CZ-133SF	モデムターミナルソフト	¥ 25,800	¥ 2?,000	現金一括払
	Z'STAFF PRO-68K	¥ 58,000	¥ 47,000	¥ 3,541×15回
	kamikaze	¥ 68,000	¥ 5?,000	¥ 3,499×18回

X1Gmodel30

X1Gの本格派セット FDD2基内蔵、専用カラーモニタはTVにも使用可能。

CZ-822C…¥118,000
CZ-820D…¥ 79,000
合計……¥197,000

特価 ¥79,800
お支払例 ¥ 7,382×12回 ¥ 5,076×18回
¥ 3,924×24回 ¥ 3,245×30回

X1turboZII

X1turboZの本格派セット。TV付2モードオートスキャンディスプレイ。

CZ-881C…¥179,800
CZ-880D…¥109,800
合計……¥289,600

特価 ¥2?4,000
お支払例 ¥20,720×12回 ¥11,013×24回
¥ 7,716×36回 ¥ 6,067×48回

X1twin

HEシステムを搭載、最上級ゲーム機とパソコンが合体。

CZ-830C…¥ 99,800
CZ-820C…¥ 79,800
合計……¥179,600

特価 ¥94,800
お支払例 ¥ 8,769×12回 ¥ 6,030×18回
¥ 4,661×24回 ¥ 3,265×36回

X1turboZIII

X1ターボシリーズの独自の機能を全継承 VCCIゼロdB基準に適合させた。

CZ-888C…¥169,800
CZ-860D…¥ 99,800
合計……¥269,600

特価 ???
応談 価格はご相談に応じます、電話でお問い合せ下さい。

●分割回数は3回〜48回まで自由に選べます。

●セットの組合せは自由!広告に出ていない他の機種はお問合せ下さい。

●頭金なし(手軽な電話クレジット)●製品先取り(お支払いは約1〜2ヶ月後から)●低金利クレジット(1回の支払いは2,700円以上で3〜48回。ボーナス併用も可)●カレッジクレジット(保証人なし。但し満20歳以上の学生の方)●18歳未満の方(ご両親が代理購入者としてお申し込み下さい)●納期(通常の場合、当社に申込書が到着後1週間以内。特に人気のある商品で品薄の場合、少々納期が遅れることがありますので御了承下さい)●完全保証(すべてメーカー保証書付。アフターケア万全)●全国代引(お届けした者に、代金をお支払いいただく方法です。但し手数料1,000円)

AM10時からPM9時まで受付 日曜・祝日も営業

安心と信頼の誌上ショッピング メディアショップ お申込みは今すぐ電話かハガキで!!

株式会社 メディアショップ ハイランド 〒239 神奈川県横須賀市ハイランド3-9-6

電話でのお申込みは

東京受付センター
☎03(252)2608

大阪受付センター
☎06(363)1605

年中無休AM10時～PM10時

ハガキでのお申込みは

〒239
神奈川県横須賀市ハイランド3-9-6
(株)メディアショップハイランド
Oh!X 係

申込書
- 商品名(商品番号)
- 支払回数
- お名前
- 生年月日
- ご住所、電話番号
- お勤め先
名称、住所、電話番号

通信販売のお申込み方法

▶現金一括でお申込みの方
- 商品名(商品番号)及び、住所、氏名、電話番号、ご覧の雑誌名をご記入の上、代金を現金書留でお送り下さい。
- 振込をご希望の方は、必ずお振込前にお電話又はおハガキで、お知らせ下さい。
〈銀行振込〉協和銀行・久里浜支店 当座No.2945
〈郵便振替〉横浜9-42177

▶クレジットでお申込みの方
- 電話かハガキでお申込み下さい。
クレジット申し込み用紙をお送り致しますので、ご記入の上、当社へお送り下さい。

SHARP X68000

X68000 オリジナルグッズ プレゼント実施中!!
- X ウォールポケット
- X ポーチ
- X マウスパット

御買上げのお客様に、上記X68000オリジナルグッズを1点、もれなくプレゼント。

SHARP パソコンテレビ X1G Model30

●CZ-822C
ミニフロッピーディスクドライブ2ドライブ内蔵。最高得点も必勝プロセスもビデオに録れる初のマルチビジュアル端子搭載。
●CZ-820D
14型カラーディスプレイテレビ。

標準価格 197,800円

商品番号 086	一括払価格 79,800円
20回	初回4,600円・4,600円×19回
30回	初回4,000円・3,200円×29回

SHARP X1 turbo ZIII

●CZ-888C
画像取り込み、ビデオ編集ステレオFM音源。多彩な機能で広がるアートワーク。
ADVANNCED TURB
●CZ-860D
14型カラーディスプレイテレビ。

標準価格 269,600円

商品番号 200	一括払価格 特別価格
24回	初回11,260円・10,600円×23回
36回	初回9,140円・7,400円×35回

SHARP X1 twin

●CZ-830C
X1twinのtwinはtwincomだ。HEシステムを内蔵し、X1シリーズ新境地を開く入門機。
●CZ-820D
14型カラーディスプレイテレビ

標準価格 179,600円

商品番号 193	一括払価格 94,800円
24回	初回5,350円・4,600円×23回
36回	初回4,850円・3,200円×35回

SHARPパーソナルワープロ

●WD-290F
手軽に枠編集が楽しめるスタンダードタイプ。約10万語48/52ドット印字

標準価格 128,000円

商品番号 207	一括払価格 99,800円
24回	初回6,600円・4,800円×23回
36回	初回4,000円・3,400円×35回

SHARPパーソナルワープロ

●WD-310F
52ドットプリンタを搭載した、高速・高品位・編集ワープロ。

標準価格 148,000円

商品番号 208	一括払価格 特別価格
24回	初回5,750円・5,600円×23回
36回	初回4,950円・3,900円×35回

SHARPパーソナルワープロ

●WD-550
FDDを一体化。実務コースとも余裕をもって応えます。

標準価格 178,000円

商品番号 209	一括払価格 特別価格
24回	初回8,450円・6,500円×23回
36回	初回5,050円・4,600円×35回

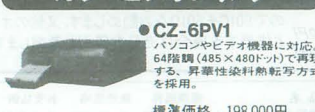
SHARPパーソナルワープロ

●WD-820
明るく見やすい、大型ディスプレイタイプ。文書の作成・管理が効率的にできる2基のFDD。

標準価格 248,000円

商品番号 210	一括払価格 148,000円
24回	初回7,560円・7,200円×23回
36回	初回7,040円・5,000円×35回

X1, X1turbo X68000シリーズ用周辺機器

カラービデオプリンタ

●CZ-6PV1
パソコンやビデオ機器に対応。64階調(485×480ドット)で再現する、昇華性染料熱転写方式を採用。

標準価格 198,000円

商品番号 149	一括払価格 特別価格
24回	初回8,850円・7,500円×23回
36回	初回8,650円・5,200円×35回

カラー イメージ スキャナー

●CZ-8NS1
高速、高精度でハイレベルな画像入力を実現。最大A4サイズの原稿をフルカラー読み取り可能。

標準価格 188,000円

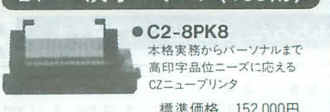
商品番号 188	一括払価格 特別価格
24回	初回7,560円・7,200円×23回
36回	初回7,040円・5,000円×35回

熱転写カラー漢字プリンタ

●CZ-8PC3
鮮やかカラー印字と高速性。ここまで身近になった24ドット熱転写カラープリンタ。

標準価格 65,800円

商品番号 191	一括払価格 特別価格
6回	初回9,620円・9,100円×5回
12回	初回5,500円・4,700円×11回

24ピン漢字プリンタ(136桁)

●C2-8PK8
本格実務からパーソナルまで高印字品位ニーズに応えるCZニュープリンタ

標準価格 152,000円

商品番号 175	一括払価格 特別価格
24回	初回5,790円・5,700円×23回
36回	初回7,410円・3,900円×35回

商品	型番	定価	価格
24ピン80桁漢字プリンタ	CZ-8PK7	¥122,000	特別価格
24ピン80桁漢字プリンタ	CZ-8PK9	¥89,800	特別価格
20MBハードディスク	CZ-620H	¥178,000	特別価格
カラーイメージユニット	CZ-6VT1	¥69,800	特別価格
スキャナ用パラレルボード	CZ-6BN1	¥29,800	特別価格
モデムユニット	CZ-8TM2	¥49,800	特別価格
1MB増設RAMボード	CZ-6BE1	¥35,000	特別価格
1MB増設RAMボード	CZ-6BE1A	¥38,000	特別価格
2MB増設RAMボード	CZ-6BE2	¥79,800	特別価格
4MB増設RAMボード	CZ-6BE4	¥138,000	特別価格
拡張I/Oボックス	CZ-6EB1	¥88,000	特別価格
ユニバーサルI/Oボード	CZ-6BU1	¥39,800	特別価格
GP-IBボード	CZ-6BG1	¥59,800	特別価格
RGBシステムチューナー	CZ-6TU	¥35,800	特別価格
数値演算プロセッサボード	CZ-6BP1	¥79,800	特別価格
FAXボード	CZ-6BC1	¥79,800	特別価格
アイテック20MBHD	ITX-203	¥125,000	特価¥88,000
アイテック40MBHD	ITX-403	¥188,800	特価¥130,000

シャープオリジナルソフトウェア

商品	型番	定価	価格
DATA PPO-68K	CZ-220BS	¥58,000	特別価格
TOP財務会計	CZ-227BS	¥200,000	特別価格
CコンパイラPRO-68K	CZ-211LS	¥39,800	特別価格
CARD PPO-68K	CZ-226BS	¥29,800	特別価格
AI-68K	CZ-234LS	¥188,000	特別価格
NEW Printshop PRO-68K	CZ-221HS	¥19,800	特別価格
OS-9/X68000	CZ-219SS	¥29,800	特別価格
Musicstudio PRO-68K	CZ-237MS	¥25,800	特別価格
MUSIC PRO-68K 〔MIDI〕	CZ-247MS		特別価格

安心と信頼 メディアショップ ハイランド

①完全保証 全国どこでもアフターケアOK
②全国無料配送 日曜配送可能
③支払回数は 予算に応じ3～36回 ボーナス併用可
④低金利クレジット 実質年率12.50～23.75%
⑤FAXでも注文OK FAX：0468(48)3273
⑥その他広告以外の商品も取扱っております。お気軽にお問合せ下さい。

価格問合せや商品説明はお問合せテレフォン ☎0468(48)3290で!

▶当社はX-68000の販売認定店です◀

パソコン・AV専門 O.A.ランド

消費税前 大特価セール

セール期間 '89 2・16→3・15

ランド特選!! SHARP X68000 ACE-HD / ACE セット

ACE-HD セット
- CZ-611C……………………定価¥399,800
- CZ-611D……………………定価¥145,000
- CZ-6ST1……………………定価¥ 5,800
- MD-2HD 20枚サービス
- 市販ゲームソフト2本サービス

他店には負けません TEL下さい 合計価格¥550,600

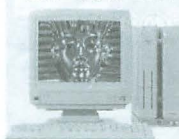
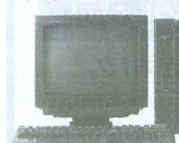
現金大特価!! 安いぞ

安すぎて 表示できません!!

ACE セット
- CZ-601C……………………定価¥319,800
- CZ-601D……………………定価¥119,800
- CZ-6ST1……………………定価¥ 5,800
- MD-2HD 20枚サービス
- 市販ゲームソフト2本サービス

OAランドで買わなきゃ損をする! 合計価格¥445,400

現金大特価!! 推選

安すぎて 表示できません!!

NEW X-1ターボZⅢセット

CRTクリーナー キーボードカバープレゼント

安すぎて ゴメンなさい!

Ⓐセット
- CZ-888CBK…定価¥169,800
- CZ-880DBK…定価¥ 99,800
- CZ-6ST1B…定価¥ 5,800 (チルトスタンド)
- MD-2HD 20枚サービス

合計定価¥275,400

現金価格 **特価中TEL下さい**

Ⓑセット
- CZ-888CBK…定価¥169,800
- CZ-830DBK…定価¥ 98,000
- CZ-6ST-1B…定価¥ 5,800 (チルトスタンド)
- MD-2HD 20枚サービス

合計価格¥273,600

合計価格 **特価中TEL下さい**

安い!! X-1ターボⅡセット 単体販売OK

Ⓐセット
- CZ-881CBK…定価¥179,800
- CZ-880DBK…定価¥109,800
- CZ-6ST1-B…定価¥ 5, 800 (チルトスタンド)
- MD-2HD 20枚サービス

合計価格¥295,400

現金価格

Ⓑセット
- CZ-881CBK…定価¥179,800
- CZ-830BK……定価¥ 98,000
- CZ-6ST1-B…定価¥ 5,800 (チルトスタンド)
- MD-2HD 20枚サービス

合計価格¥283,600

現金価格

一度TEL下さい X-1Gセット 単体販売OK

X-1Gセット お買得
- CZ-822CB……定価¥118,000
- CZ-820DB……定価¥ 79,800
- MD-2D 20枚サービス

合計価格¥197,800

① X-1Gセット ＋パソコンラック
現金特価**¥81,000**

② X-1Gセット ＋市販ゲームソフト2本
現金特価**¥79,800**

周辺機器コーナー

X1用	定価	特価	X68000用	定価	特価
CZ-8BV2	¥ 39,800	¥ 31,000	CZ-6PU1A	¥ 38,000	¥ 30,000
CZ-8BR1	¥ 29,800	¥ 23,000	CZ-6BM1	¥ 26,800	¥ 21,000
CZ-8DT2	¥ 44,800	¥ 35,000	CZ-6BE1	¥ 88,000	¥ 69,800
CZ-8BS1	¥ 23,800	TEL下さい	CZ-6VT1	¥ 69,800	TEL下さい
CZ-8TM2	¥ 49,800	¥ 38,000	CZ-8NS1	¥188,000	¥149,000
CZ-8EB3	¥ 33,800	¥ 27,000	CZ-6BC1	¥ 79,800	¥ 63,000

プリンターセットコーナー

	定価	特価
①CZ-6PU1(カラービデオプリンター)	¥198,000	¥152,000
②CZ-8PC3(カラープリンター)	¥ 65,800	¥ 53,000
③CZ-8PK8(ドットプリンター)	¥152,000	¥115,000
④CZ-8PK7(ドットプリンター)	¥122,000	¥ 93,000
⑤PC-PR201TH(カラープリンター)	¥145,000	¥103,000
⑥PC-PR201G(ドットプリンター)	¥158,000	¥ 99,000

その他、周返機器・プリンター ソフトウェアー 20%～25% OFF!!

X68000用ソフトウェアー・コーナー

	定価	特価
①CZ-212BS(BUSINESS)	¥ 68,000	¥ 53,000
②CZ-220BS(DATA)	¥ 58,000	¥ 45,000
③CZ-215MS(Sampling)	¥ 17,800	¥ 13,800
④CZ-221HS(NEW Print Shop)	¥ 10,800	¥ 15.500
⑤CZ-227BS(TOP財務会計)	¥200,000	¥158,000
⑥CZ-226BS(CARD)	¥229,800	¥ 23,000
⑦CZ-223CS(Communication)	¥ 19,800	¥115,500
⑧CZ-213MS(MUSIC)	¥ 18,800	¥ 14,800
⑨CZ-211LS(C compiler)	¥ 39,800	¥ 31,000
⑩C-TRACE(キャスト)	¥ 68,000	¥ 52,000
⑪EW(イースト)	¥ 38,000	¥ 29,000

■ハードディスク ■特価品もありますのでTEL下さい。

- アイテック IT-MJ4(I/F付)……特価¥98,000
- アイテック IT-MJ4 C(I/F付)……特価¥109,000
- ウィンテック HD-404HS(I/F付)……特価¥108,000
- コンピュータ CRC-HD4A(I/F付)……特価¥89,000
- スナイパー SP-340(I/F付)……特価¥92,000
- アイテック ITH-320S(I/F付)……特価¥79,800
- ウィンテック HD-202(I/F付)……特価¥58,000
- スナイパー SR-520(I/F付)……特価¥55,000
- コンピュータ CRC-HD2A(I/F付)……特価¥62,000
- ロジテック LHD-32NR(I/F付)……特価¥80,000

今月の特価品 各一台限り

■A紙品(美品・POP品) ■B級品(キズ少々) ■C級品(キズ有り)

		A級品	B級品	C級品
X68000シリーズ	CZ-611C	¥282,000より	¥279,000	¥269,000
	CZ-601C	¥215,000より	¥210,000	¥205,000
	CZ-611D	¥104,000より	¥ 99,000	¥ 85,000
X-1ターボシリーズ	CZ-881	¥ 88,000より	¥ 85,000	¥ 83,000
MZシリーズ	MZ-2861	¥198,000より	¥185,000	¥158,000
その他	CZ-8NS1	¥145,000より	¥135,000	¥128,000

その他、いろいろありますのでTEL下さい!!

新年ディスケットセール

■D社クリーニングディスク付
- MD-2D 10枚 ¥ 850
- MD-2DD 10枚 ¥1,150
- MD-2HD 10枚 ¥1,180
- MF-2D 10枚 ¥2,600
- MF-2DD 10枚 ¥2,700
- MF-2HD 10枚 ¥4,200

中古パソコン(価格・在庫は変動します。予約は5日以内といたします。)

機種	価格	機種	価格
PC-9801VX21	¥235,000より	PC-8801mkII 30	¥ 35,000より
PC-9801VX2	¥210,000より	PC-8801mkII SR	¥ 73,000より
PC-9801VM2	¥170,000より	PC-8801mkII FR30	¥ 68,000より
PC-9801VF2	¥118,000より	PC-8801mkII MR	¥ 88,000より
PC-9801M2	¥145,000より	PC-88VA	¥148,000より
PC-9801F2	¥ 88,000より	PC-8801mkII FH30	¥ 85,000より
PC-9801UV21	¥148,000より	PC-8801FA	¥108,000より
PC-98LTM1(640KB)	¥ 89,000より	X-1Gモデル30	¥ 25,000より
PC-286モデル0	¥168,000より	X-1ターボII	¥ 68,000より
		FM-77D2	¥ 28,000より
PC-286V-STD	¥202,000より	FM-77AV2	¥ 42,000より
X-68000	¥188,000より	FM-77AV20	¥ 52,000より

通信販売のご案内

—全国通販—

■銀行振込で申し込みの方は商品名及びお客様の住所・氏名・電話番号をお知らせ下さい。

〔振込先〕第一勧業銀行 渋谷支店
普通No.1163457 (株)オーエーランド

■現金書留で送金されるお客様は電話番号と商品名、数量を明記して同封して下さい。■クレジットでご購入を希望される方は申し込み用紙をお送り致しますのでご記入の上返送して下さい。20才以上の方は、原則として保証人不要です。クレジットは1～60回払で月々5,000円よりご自由に設定できます。

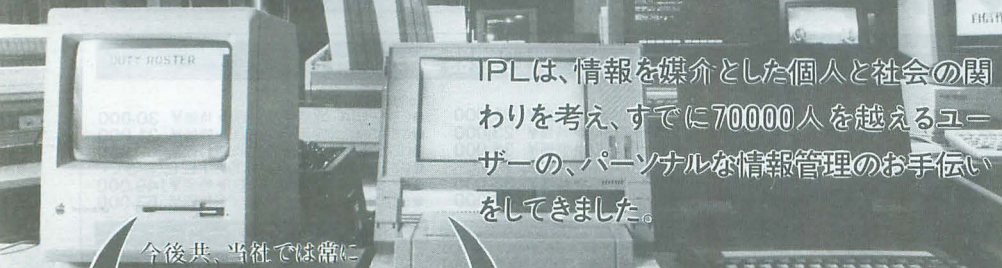
品川,目黒方面 首都高速3号線 O.A.ランド 渋谷駅 井の頭線渋谷駅 道玄坂 ヤマハ J&P 109 神泉駅 明治通り 山手線 東急百貨店 西武百貨店

- 下取・買取は電話で見積りしております。責任を持って下取りさせて頂きます。
- ご注文、お問合せは…毎日午前10時から午後7時まで
- 商品のお届けは…入金確認後、即日発送致します。

㈱オー・エー・ランド

〒150東京都渋谷区円山町20-4 第5日新ビル1F

☎(03)770-8855

関東エリアの送料は、1個につき¥1,000です。

今後共、当社では常に"真のサービスとは何か"を基本ポリシーに信頼性、経済性の高い会社として、多様なニーズに応え続けたいと考えています。

70,000人もの人々が体感した安心感。――信頼のIPLワイドサポート

●業界初、IPLでこそ成し得た3倍保証。

メーカー保証12ヶ月の商品なら36ヶ月の保証と長期間の保証を実施。末長く安心してご利用いただけるよう、IPLが成し得たワイドなサポート体制。

●IPLだからこそ初期不良への保証も万全。交換期間も1ヶ月ともっとも長期間です。IPLだからこそ安心が長続きします。

比べてほしいから、ご紹介します。――さらにお買得IPLクレジット

●ステップアップクレジットがおトク。

まず月々1,000円からスタートして2年後から3,000円アップ。ボーナスも1年後1万円。3年後3万円。また夏のボーナスを貯金して冬のボーナスからのお支払いも大丈夫。夏・冬のボーナスどちらか一つをセレクト。ボーナス年一回だけもOK。システムはすぐお手元へ。冬のボーナス一括、冬夏ボーナス2括払いもOK。

●追加購入もクレジットだから便利。

追加購入も買い換えもご利用中のIPLクレジットを月々僅か1,000円ずつの調整でOK。

●IPLは1988年8月21日ついに70,001人めのお客様を迎えました。

ORDER TELEPHONE

電話受付:AM10:00～PM8:00 水曜定休日

本社 0467-24-7511
大阪 06-311-2736

銀座 03-541-3058　青山 03-470-0061　札幌 011-621-1444
仙台 022-266-0531　広島 082-293-7881　福岡 092-481-2644

商品管理部(納期、配達日のお問合せ、ご指定日のご連絡) 0467-24-1154
メンテナンス部(ハード上のご相談、お問合せ、初期不良の対応) 0467-24-0453
FAX(ご注文、お見積り、カタログ編集などスピーディに) 0467-24-0561
タイムリーボックス(ホットな新製品ニュースをお知らせします。) 0467-24-0941

ご注文お問合せ ――0467-24-1154
下取りホットライン ――0467-24-2040

アクセス No.T0331

価格¥31,400 ➡ IPL超特価

品名	価格
Super大戦略68K	¥ 8,800
めぞん一刻 完結	¥ 7,800
三国志	¥ 14,800

標準価格¥31,400

月々	ボーナス
¥1,400×24回	なし
¥ 2,700×12回	なし

アクセス No.T0333

価格¥214,800 ➡ IPL超特価

品名	価格
EW & E1(漢字変換フロントプロセッサ搭載、高速日本語ワープロ)	¥ 38,000
CZ-212BS(データベース表グラフ、ソート機能、斜線、横倍角、網掛け下線)	¥ 68,000
VP-135EX(抜群のコストパフォーマンスプルトラクタ付136桁24ドット)	¥100,000
#8226	¥ 8,800

標準価格¥214,800

月々	ボーナス
¥2,800×72回	なし
¥ 5,100×36回	なし

◆全国無料配達◆　◆日曜・祭日指定配達◆

アクセス No.T0337

価格¥126,000 ➡ IPL超特価

品名	価格
Z'sSTAFF PRO 68K(グラフィックツール)	¥ 58,000
C-TRACE 68(X68000、3次元グラフィック!多彩な図形を作る)	¥ 68,000

標準価格¥126,000

月々	ボーナス
¥2,400×48回	なし
¥ 3,100×36回	なし

SHARP X-68000

アクセス No.X0368

価格¥718,600 ➡ IPL超特価

品名	価格
CZ-601C(CPU68000、2Mバイト, 65536同時発色)	¥319,800
CZ-601D(0.39ミリ、アナログ3モードオートスャン)	¥119,800
CZ-6ST(角度自由自在、調節OK!)	¥ 5,800
3M ブランクディスケット(5"2HD*10枚)	¥ 24,000
CZ-8PC3(10"カラー熱転写、ハガキ可。漢字53字/秒)	¥ 65,800
CZ-6BM1(MIDIボード)	¥ 26,800
CZ-6BP1(数値演算プロセッサボード)	¥ 79,800
C-TRACE(CGアニメーションソフト)	¥ 68,000
CZ-218AS(沙羅曼蛇)	¥ 8,800
☆IPL's 5 BIG SUPPORT	¥ 0

標準価格¥718,600

月々	ボーナス
¥3,000×72回	3.75万×12回
¥ 5,000×72回	2.55万×12回
¥ 5,000×60回	3.43万×10回
¥ 7,900×48回	3.0万×8回
¥10,000×36回	3.96万×6回

アクセス No.X0371

価格¥511,400 ➡ IPL超特価

品名	価格
CZ-601C(CPU68000、2Mバイト, 65536同時発色)	¥319,800
CZ-601D(0.39ミリ、アナログ3モードオートスャン)	¥119,800
CZ-6ST1(角度自由自在、調節OK!)	¥ 5,800
3M ブランクディスケット(5"2HD*10枚)	¥ 24,000
源平討魔伝	¥ 7,800
信長の野望全国版	¥ 9,800
スペースハリアー	¥ 6,800
CZ-218AS(沙羅曼蛇)	¥ 8,800
ドラゴンスピリット	¥ 8,800
☆IPL's 5 BIG SUPPORT	¥ 0

標準価格¥511,400

月々	ボーナス
¥3,000×72回	2.0万×12回
¥ 4,000×60回	2.0万×10回
¥ 4,600×48回	2.5万×8回
¥ 6,400×36回	3.0万×6回
¥10,000×24回	3.9万×4回

アクセス No.X0372

価格¥682,500 ➡ IPL超特価

品名	価格
CZ-601C(CPU68000、2Mバイト, 65536同時発色)	¥319,800
CZ-601D(0.39ミリ、アナログ3モードオートスャン)	¥119,800
CZ-6ST1(角度自由自在、調節OK!)	¥ 5,800
3M ブランクディスケット(5"2HD*10枚)	¥ 24,000
CZ-213MS(MUSIC PRO 68K)	¥ 18,800
CZ-214MS(SOUND PRO 68K)	¥ 15,800
CZ-226BS(ワープロ機能を備えたカード型リレーショナルデータベース)	¥ 29,800
CZ-8PC3(10"カラー熱転写、ハガキ可。漢字53字/秒)	¥ 65,800
CZ-6SDI(X68専用キャスター, スライドテーブル付キーボード収納OK)	¥ 44,800
SNC-081(布張肘掛け付き回転イス)	¥ 20,500
信長の野望郡雄伝	¥ 9,800
殺人倶楽部	¥ 7,800
☆IPL's 5 BIG SUPPORT	¥ 0

標準価格¥682,500

月々	ボーナス
¥3,900×72回	3.0万×12回
¥ 5,000×60回	3.15万×10回
¥ 6,000×48回	3.8万×8回
¥10,000×36回	3.53万×6回
¥14,800×24回	5.0万×4回

超 低 金 利 ………　組 合 せ 自 由　全 国 無 料 配 送

アクセス No.X0369

価格¥603,200 ➡ IPL超特価

品名	価格
CZ-601C(CPU68000、2Mバイト、65536同時発色)	¥319,800
CU-21CD(迫力の21"カラーアナログCRT3モードマルチスキャン方式)	¥139,800
CZ-213MS(MUSIC PRO 68K)	¥ 18,800
CZ-214MS(SOUND PRO 68K)	¥ 15,800
CZ-215MS(AD PCM機能をサポートしたサンプリングエディタ)	¥ 17,800
CZ-8PC3(10"カラー熱転写、ハガキ可、漢字53字/秒)	¥ 65,800
めぞん一刻 完結編	¥ 7,800
ドラゴンスピリット	¥ 8,800
Super大戦略68K	¥ 8,800
☆IPL's 5 BIG SUPPORT	¥ 0

¥2,900 ×72回 ボーナス 3.0万×12回　標準価格¥603,200

月額	ボーナス
¥ 5,000×60回	ボーナス 2.45万×10回
¥ 6,000×48回	ボーナス 3.0万×8回
¥ 9,100×36回	ボーナス 3.0万×6回
¥ 7,900×72回	ボーナス なし

アクセス No.X0370

価格¥458,000 ➡ IPL超特価

品名	価格
CZ-601C(CPU68000、2Mバイト、65536同時発色)	¥319,800
CZ-603D(超高解像度0.31ドット14"チルト台付)	¥ 84,800
3M ブランクディスケット(5"2HD*10枚)	¥ 24,000
Super大戦略68K	¥ 8,800
大海令	¥ 12,800
源平討魔伝	¥ 7,800
☆IPL's 5 BIG SUPPORT	¥ 0

¥3,100 ×60回 ボーナス 2.0万×10回　標準価格¥458,000

月額	ボーナス
¥ 5,600×72回	ボーナス なし
¥ 4,400×48回	ボーナス 2.0万×8回
¥ 5,000×36回	ボーナス 3.0万×6回
¥ 9,600×24回	ボーナス 3.0万×4回

こんなにかかる修理費用

プリンタヘッド交換¥29,500以上/98シリーズメインボード交換¥21,600以上/ドライブ交換¥13,200以上

SHARP X68000 ACE HD

アクセス No.X0377

価格¥1,055,800 ➡ IPL超特価

品名	価格
CZ-611C(20MHDD搭載、65536色発色、FM8音源内蔵)	¥399,800
CZ-601D(0.39ミリ、アナログ3モードオートスキャン)	¥119,800
CZ-6ST1(角度自由自在、調節OK!)	¥ 5,800
3M ブランクディスケット(5"2HD*10枚)	¥ 24,000
CZ-6BM1(MIDIボード)	¥ 26,800
CZ-237MS(ミュージックスタジオ MIDI楽器演奏が楽しめる)	¥ 25,800
CZ-6BP1(数値演算プロセッサボード)	¥ 79,800
CZ-219SS(OS/9登場マルチメディア、マルチタスク、リアルタイム機能)	¥ 29,800
CZ-211LS(C compilerソフト開発を効率良(サポート))	¥ 39,800
C&ProfessionalPack(OS-9開発ツール)	¥ 58,000
CZ-213MS(MUSIC PRO 68K)	¥ 18,800
CZ-214MS(SOUND PRO 68K)	¥ 15,800
CZ-215MS(AD PCM機能をサポートしたサンプリングエディタ)	¥ 17,800
C-TRACE(CGアニメーションソフト)	¥ 68,000
CZ-6BE1A(1MB増設RAMボード)	¥ 38,000
CZ-6EB1(拡張I/Oボックス)	¥ 88,000
☆IPL's 5 BIG SUPPORT	¥ 0

¥5,800 ×72回 ボーナス 5.0万×12回　標準価格¥1,055,800

月額	ボーナス
¥ 9,900×72回	ボーナス 2.5万×12回
¥10,000×60回	ボーナス 3.8万×10回
¥ 11,300×48回	ボーナス 5.0万×8回
¥16,900×36回	ボーナス 5.0万×6回

アクセス No.X0373

価格¥806,670 ➡ IPL超特価

品名	価格
CZ-611C(20MHDD搭載、65536色発色、FM8音源内蔵)	¥399,800
CZ-611D(0.31ミリ、アナログ3モードオートスキャン)	¥145,000
3M ブランクディスケット(5"2HD*10枚)	¥ 24,000
CZ-6ST1(角度自由自在、調節OK!)	¥ 5,800
C-TRACE(CGアニメーションソフト)	¥ 68,000
Z's STAFF PRO 68K(グラフィックツール)	¥ 58,000
CZ-211LS(C compilerソフト開発を効率良(サポート))	¥ 39,800
CZ-8PC3(10"カラー熱転写、ハガキ可、漢字53字/秒)	¥ 65,800
A4カット紙(100枚)	¥ 470
☆IPL's 5 BIG SUPPORT	¥ 0

¥3,000 ×72回 ボーナス 4.37万×12回　標準価格¥806,670

月額	ボーナス
¥ 5,000×72回	ボーナス 3.17万×12回
¥ 7,000×60回	ボーナス 3.0万×10回
¥ 8,000×48回	ボーナス 3.8万×8回
¥10,200×36回	ボーナス 5.0万×6回

アクセス No.X0379

価格¥751,580 ➡ IPL超特価

品名	価格
CZ-611C(20MHDD搭載、65536色発色、FM8音源内蔵)	¥399,800
CZ-603D(超解像度0.31ドット14"チルト台付)	¥ 84,800
3M ブランクディスケット(5"2H*10枚)	¥ 24,000
CZ-6BE1A(1MB増設RAMボード)	¥ 38,000
CZ-211LS(C compilerソフト開発を効率良(サポート))	¥ 39,800
CZ-212BS(データベース表グラフ、ソート機能、斜線、横倍角、網掛け下線)	¥ 68,000
CZ-213MS(MUSIC PRO 68K)	¥ 18,800
CZ-214MS(SOUND PRO 68K)	¥ 15,800
CZ-237MS(ミュージックスタジオ MIDI楽器演奏が楽しめる)	¥ 25,800
CZ-224LS(ソフトウエアー開発ツール/THE福袋)	¥ 9,980
CZ-6BM1(MIDIボード)	¥ 26,800
☆IPL's 5 BIG SUPPORT	¥ 0

¥4,700 ×72回 ボーナス 3.0万×12回　標準価格¥751,580

月額	ボーナス
¥ 5,000×60回	ボーナス 3.73万×10回
¥ 8,500×48回	ボーナス 3.0万×8回
¥10,000×36回	ボーナス 4.42万×6回
¥ 9,700×72回	ボーナス なし

アクセス No.X0376

価格¥1,168,800 ➡ IPL超特価

品名	価格
CZ-611C(20MHDD搭載、65536色発色、FM8音源内蔵)	¥399,800
CZ-601D(0.39ミリ、アナログ3モードオートスキャン)	¥119,800
CZ-6ST1(角度自由自在、調節OK!)	¥ 5,800
3M ブランクディスケット(5"2HD*10枚)	¥ 24,000
CZ-6VT1(カラーイメージユニット、テロッパー機能付き)	¥ 69,800
CZ-6PV1(カラービデオプリンタ)	¥198,000
GZY50(カラービデオプリンタ用シートインクセット)	プレゼント中
CZ-8NS1(フルカラーA4ズーム機能色ずれの少ない線順次方式ソフト付き■)	¥188,000
CZ-6BN1(高速伝送、付属のソフトで約7倍UP2枚まで実装可ケーブル付)	¥ 29,800
C-TRACE 68(X68000、3次元グラフィック!多彩な図形を作る。)	¥ 68,000
CZ-8PC3(10"カラー熱転写、ハガキ可、漢字53字/秒)	¥ 65,800
☆IPL's 5 BIG SUPPORT	¥ 0

¥6,700 ×72回 ボーナス 5.0万×12回　標準価格¥1,168,800

月額	ボーナス
¥10,000×72回	ボーナス 3.0万×12回
¥10,000×60回	ボーナス 4.42万×10回
¥12,500×48回	ボーナス 5.0万×8回
¥18,500×36回	ボーナス 5.0万×6回

ビギナーズ・ホットライン

初心者の方々のために、常設の無料相談窓口を設けました。お気軽にご利用ください。

0467-24-0941

安心の3倍保証

★IPL's 5 BIG SUPPORT

1. IPL保証書付き安心の3倍保証
2. ワイドに1ヶ月間の初期不良交換サービス
3. ひとりひとりをしっかりフォローする添削付通信講座(無料)
4. キーボードレッスン添付(PC98、EPSON286シリーズ)
5. SHARP製品をシステムでお求めの方にモデム(SR30ケーブル付)をプレゼント中

アクセス No.X0378

価格¥938,200 ➡ IPL超特価

品名	価格
CZ-611C(20MHDD搭載、65536色発色、FM8音源内蔵)	¥399,800
CZ-611D(0.31ミリ、アナログ3モードオートスキャン)	¥145,000
3M ブランクディスケット(5"2HD*10枚)	¥ 24,000
CZ-6ST1(角度自由自在、調節OK!)	¥ 5,800
CZ-6BE1A(1MB増設RAMボード)	¥ 38,000
CZ-6BE4(4MB増設RAMボード)	¥138,000
CZ-6BP1(数値演算プロセッサボード)	¥ 79,800
C-TRACE(CGアニメーションソフト)	¥ 68,000
CZ-211LS(C compilerソフト開発を効率良(サポート))	¥ 39,800
☆IPL's 5 BIG SUPPORT	¥ 0

¥3,900 ×72回 ボーナス 5.0万×12回　標準価格¥938,200

月額	ボーナス
¥ 7,000×72回	ボーナス 3.1万×12回
¥ 7,800×60回	ボーナス 3.8万×10回
¥10,000×48回	ボーナス 4.15万×8回
¥13,500×36回	ボーナス 5.0万×6回

アクセス No.X0375

価格¥768,470 ➡ IPL超特価

品名	価格
CZ-611C(20MHDD搭載、65536色発色、FM8音源内蔵)	¥399,800
CZ-611D(0.31ミリ、アナログ3モードオートスキャン)	¥145,000
3M ブランクディスケット(5"2HD*10枚)	¥ 24,000
CZ-6ST1(角度自由自在、調節OK!)	¥ 5,800
CZ-211LS(C compilerソフト開発を効率良(サポート))	¥ 39,800
CZ-219SS(OS/9登場マルチメディア、マルチタスク、リアルタイム機能)	¥ 29,800
C&ProfessionalPack(OS-9開発ツール)	¥ 58,000
CZ-8PC3(10"カラー熱転写、ハガキ可、漢字53字/秒)	¥ 65,800
A4カット紙(100枚)	¥ 470
☆IPL's 5 BIG SUPPORT	¥ 0

¥3,000 ×72回 ボーナス 4.07万×12回　標準価格¥768,470

月額	ボーナス
¥ 4,800×72回	ボーナス 3.0万×12回
¥ 5,500×60回	ボーナス 3.5万×10回
¥ 7,800×48回	ボーナス 3.5万×8回
¥10,000×36回	ボーナス 4.53万×6回

アクセス No.T0330

価格¥85,600 ➡ IPL超特価

品名	価格
CZ-8PC3(10"カラー熱転写、ハガキ可、漢字53字/秒)	¥ 65,800
CZ-221HS(NEW printShop様々なカードなどを自由に作成)	¥ 19,800

¥2,300 ×36回 ボーナス なし　標準価格¥85,600

月額	ボーナス
¥ 3,300×24回	ボーナス なし

日曜・祭日・指定日配送OK!　輸送上のトラブルにも対応　お申し込みはナンバーでお願いします。

※IPL超特価システムはさらにおトクなシステム。お電話でお問合せください。

オクトで始まるパソコンワールド

☎ 03-730-6271

●営業時間 AM 11:00～9:00/日曜・祭日PM7:00● 定休日：毎週火曜日
祭日の場合は翌日になります

〒144 東京都大田区蒲田4-6-7 FAX 03-730-6273

店頭セール実施中

全国通販

電話一本で、ハイ即納

OCT-1 システム インフォメーション
- ▶全商品保証付(メーカー保証)
- ▶超低金利ハッピークレジット(1回～60回)頭金ナシOK！
- ▶ボーナス一括払いOK！ボーナス2回払いOK！！
- ▶配達日の指定OK！(万全なサポート体制)
- ▶商品の組合せ自由！ オクトフリーダムシステム
- ▶店頭デモンストレーション実施中

オクト
セレクテッドシステム

広告掲載商品以外の
製品も取扱っております。

オクトで始まるパソコンワールド。

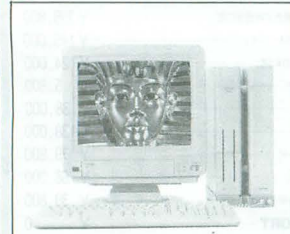
2/15～4/15まで店頭にて、ゲームソフト・オール25%OFF!!

OCT-1 蒲田 OPEN

オクト恒例 大感謝祭セール!!
X68000フェア開催中!!

《プレゼント実施中》 ★ドラゴンスピリッツ(¥8,800) ★3M ブランクディスク(MD-2HD 10枚)

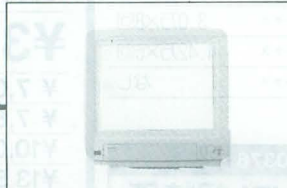
お好みのセットをお選び下さい。
送料無料

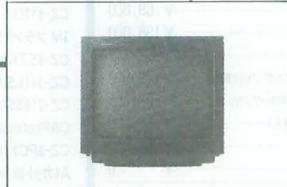
20MBハードディスク・モデル

X68000 ACE-HD
CZ-611C-GY/BK
定価¥399,800
現金特価!!
お電話下さい。
推選

ハイコストパフォーマンス
FDモデル

X68000 ACE
CZ-601C-GY/BK
定価¥319,800
現金特価!!
お電話下さい。
推選

CZ-611D-GY/BK
定価¥145,000

15型カラーディスプレイTV

CZ-601D-GY/BK
定価¥119,800

14型カラーディスプレー

CZ-603D-GY/BK
定価¥84,800

21型カラーディスプレイ

CU-21CD
定価¥139,800

ACE-HD
①CZ-611C+CZ-611D
合計金額¥544,800……………大特価TEL下さい。

全商品保証付(メーカー保証)

ACE
②CZ-601C+CZ611D
合計金額¥464,800……………大特価TEL下さい。

超低金利ハッピークレジット(1回～60回)頭金なしOK!!

ACE-HD
③CZ-611C+CZ-601D
合計金額¥519,600……………大特価TEL下さい。

ボーナス一括払いOK！ボーナス2回払いOK！

ACE
④CZ-601C+CZ-601D
合計金額¥439,600……………大特価TEL下さい。

配達日の指定OK！(万全なサポート体制)

ACE-HD
⑤CZ-611C+CZ-603D
合計金額¥484,600……………大特価TEL下さい。

商品の組合せ自由！オクトフリーダムシステム

ACE
⑥CZ-601C+CZ-603D
合計金額¥404,600……………大特価TEL下さい。

店頭デモンストレーション実施中！

ACE-HD
⑦CZ-611C+CU-21CD
合計金額¥539,600……………超特価TEL下さい。

オクト特選セット…電話は最後に

ACE
⑧CZ-601C+CU-21CD
合計金額¥459,600……………超特価TEL下さい。

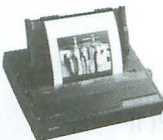
オクト推奨セット…電話はお早目に

※クレジットの回数は1回～60回、ボーナス併用などありますのでお電話でお問合せ下さい。

■本体セット：送料¥2,000 ●店頭デモ実施中…専門の係員が詳細にアドバイス致します。ぜひご来店下さい。

ご来店で、お買い上げの方には粗品プレゼント

■特に人気のある商品によっては、しばらくお待ち願うことがありますのでご了承下さい。

厳選された製品を、より安く、より早く、皆様のお手元に!!

広告掲載商品以外の製品も取扱っております。

X68000周辺機器大セール実施中 全商品送料無料

パソコンチューナー
AN-8TV(定価35,800)

●CZ-603D・CU-21CDでTVが見れる。
●ビデオ入力端子
●モニター出力端子
●スーパーインポーズ表示可能

大特価 CZ-603D・CU-21との組合せでドーン!!!

カラーイメージスキャナ
CZ-8NS1(定価¥188,000)

●A-4サイズまでの写真・図形フルカラー読み取り ●縮少・拡大自在

大特価 興味のある方TEL下さい

カラービデオプリンタ
CZ-6PV1(定価¥198,000)

●オリジナルCGや取り込んだ画像を色鮮やかにコピー!!

大特価 ビデオと組合せると最高!!

カラーイメージユニット
CZ-6VT1-BK(定価¥69,800)

●イメージ豊かなアートワークをサポート!!おしゃれなアートが楽しめます。

大特価 アート豊かな感性の方へ。

型名	商品	定価	特価	型名	商品	定価	特価
CZ-6BEI	IBM増設RAMボード	¥38,000	大特価	CZ-6EBI	拡張I/Oボックス	¥88,000	大特価
CZ-6BE2	2MB増設RAMボード	¥79,800	大特価	CZ-8TMZ	モデムユニット	¥49,800	大特価
CZ-6BGI	GP-IBボード	¥59,800	大特価	CZ-6BN-I	スキャナ用パラレルボード	¥29,800	大特価
CZ-6BPI	プロセッサ・ボード	¥79,800	大特価	CZ-8NTI	トラックボール	¥13,800	大特価
CZ-6BCI	FAXボード	¥79,800	大特価	AN-160SP	アンプ内蔵スピーカー	¥59,800	大特価
CZ-6BMI	MIDボード	¥26,800	大特価	CZ-6BUI	ユニバーサルI/Oボード	¥39,800	大特価

熱転写カラー漢字プリンター 用紙プレゼント 送料無料
CZ-8PC3 ¥65,800

●24ドットサーマルヘッド
●B5~B4縦まで
●ハガキ可能
●カラーハードコピー可能

大特価 オクト推選TEL下さい!

①CZ-8PK7(24ピン80桁)
定価¥122,000
大特価・TEL下さい。

②CZ-8PK8(24ピン136桁)
定価¥152,000
大特価・TEL下さい。

③CZ-8PK9
定価¥89,800
大特価・TEL下さい。

パソコンラック(4段) 送料無料
推奨

キミだけのパソコン・スペースを作っちゃおう!!
移動自由自在
サイズ
1245(H)×614(D)×600(W)m/m
定価¥22,800
大特価¥12,000

X68000ソフト大セール実施中 ※ゲームソフトオール23%off(送料無料)

〈グラフィック〉●Z's STAFF PRO68K Ver.2.0
(シャフト)定価¥58,000
オクト特価**¥41,000**

〈データベース〉●KAMIKAZE
(サムシンググッド)定価68,000
オクト特価**¥47,000**

〈グラフィック〉●C-TRACE68
(キャスト)定価¥68,000
オクト特価**¥51,000**

〈C言語〉●C&Professional Pack
(マイクロウェア ジャパン)定価¥58,000
オクト特価**¥44,800**

型名	商品	定価	特価
BUSINESS PRO68K	統合型表計算	¥68,000	大特価
CARD PRO68K	カード型データベース	¥29,800	大特価
DATA PRO68K	コマンド型データベース	¥58,000	大特価
COMMUNICATION PRO68K	通信ソフト	¥19,800	大特価
OS-9 X68000	マルチタイム リアルタイム オペレーティング システム	¥29,800	大特価
MUSIC PRO68K	楽譜ワープロ	¥18,800	大特価
SOUND PRO68K	サウンドエディタ	¥15,800	大特価
NEW PRINT SHOP PRO68K	ポップアートツール	¥19,800	大特価
C-COMPILER PRO68K	Cコンパイラ	¥39,800	大特価
EW	ワープロ	¥38,000	¥29,800
G-68	グラフィックツール	¥14,800	¥12,000
E-68K	スプライトエディタ	¥19,800	¥16,000

店頭ゲームソフトオール23%off!ビジネスソフト23%より特価中

★通信販売お申込みのご案内★ 〒144 東京都大田区蒲田4-6-7 TEL:03-730-6271

お申込みはお電話でお願いします。お客様の〈住所〉〈氏名〉〈電話番号〉及び〈商品名〉をお知らせ下さい。●入金確認後ただちに商品をご送付いたします。

現金一括払い
銀行振込:お近くの銀行より(電信扱い)にてお振込み下さい。
現金書留:封筒の中に住所・氏名・商品名をご記入の上当社までお送り下さい。

クレジット
専用お申込用紙をお送り致します。ので、必要事項をご記入、ご捺印の上ご返送下さい。手続きは簡単です。

振込先
三菱銀行 蒲田支店 ㊂No.0278691
東京都民銀行 蒲田支店 ㊂No.0320955
株式会社 億人(オクト)

※掲載の価格は1/20 現在ですので、まずは、お電話にてご確認ください。

オクトで始まるパソコンワールド。

ご来店で、お買い上げの方には粗品プレゼント

SHARP
パーソナルコンピュータMZ-2861

これ一台で、アレコレできます。

高性能ワープロ+高性能パソコン

●日本語ワープロ「書院28」搭載!
●MS-DOS™V3.1標準装備!

16ビットパーソナルコンピュータ

MZ-2861

MZ-2861 標準価格328,000円

MZ-2531 V.2 BASIC +通信ソフト プレゼント '89・2月28日まで

超特価!!!

下取りセールもOKです。

タイプ、消耗程度により査定致しますので、詳しくは電話でお問い合わせください。(0426-45-3001~3)

特典2大プレゼント!!
お買い上げの方全員にもれなく、「MZ-2800コンプリートガイドブック」定価¥1,300(BNN第2企画部編)と「パソコン使ってますます便利・ファクシミリ活用法」定価¥1,800(藤本考一郎著)の2冊を進呈いたします。

OAソフトウェアも大充実!

UPシリーズ

MZ-2861の日本語入力機能を有機的に活かす統合OAソフトウェア「UPシリーズ」。ディスクパブリッシングという新しいジャンルのレイアウトワープロ、集計表・グラフ作成統合ソフトウェア、自由度の高いカード型データベース、アウトラインプロセッサというジャンルの新しい企画書作成ソフトウェア。オフィスワークを代表的な4つの局面からアプローチして専門化した、切れモノOAツールです。

日本語レイアウトワープロ■デスクUP(1P-1251)
定価¥88,000➡特価¥70,400
集計表・グラフ作成ソフト■チャートUP(1P-1252)
定価¥55,000➡特価¥44,000
カード型データベース■UPクリッパー(1P-1253)
定価¥77,000➡特価¥61,600
企画書作成ソフト■プランUP(1P-1254)
定価¥66,000➡特価¥52,800

MZ-2861の多彩な周辺機器

●MZ-1D26(14型カラーディスプレイ)……¥89,800
●MZ-1D27(15型カラーディスプレイTV付)……¥127,000
●Cu-14BD(14型カラーディスプレイAN1508付)……¥66,000
●SS-SC28+WD-05HS(スキャナセット)……¥79,600
●SS-SC28M(モノクロハンディスキャナ)……¥49,800
●MZ-1P23(レーザープリンター)……¥950,000
●MZ-1P27(水平インサートプリンタ)……¥268,000
●MZ-1P28(80桁漢字プリンタ)……¥148,000
●MZ-1P29(136桁漢字プリンタ)……¥168,000
●IO-730(136桁インクジェットプリンタ)……¥230,000
●MZ-1R35(1MBRAM)……¥55,000
●MZ-1R36(1MB増設RAM)……¥45,000
●MZ-1V01(イメージ情報ステーション)……¥278,000
●IP-1256(パソコンFAX28)……¥99,800
●MZ-1X29(マウス)……¥13,800
●MZ-1X30(モデムホン)……¥98,000
●MZ-1F23(20MBハードディスク)……¥29,800
●MZ-1E35(ADPCMボード)……¥49,800
●MZ-1E39(RS232C/2ch)ボード……¥39,800

※表示の価格は定価につき、割引価格はお問い合せください。

☎0426-45-3001~3
FAX.0426-44-6002
●営業時間/10:00~19:00●電話受付/20:00迄可●定休日/日曜日(祭日営業)
SHARP SUPER XEX SHOP
アイビット電子株式会社 〒192 東京都八王子市北野町560-5

X1・MZ周辺機器他、シャープ製品徹底の品揃え

もちろん本体新製品から他店では入手しにくい旧タイプ周辺機器まで全品新品保証付。しかも大特価徹底の品揃え。特にひとつ前のタイプは絶対のお買い得です。(旧タイプは限定数のため、電話で在庫をお確かめの上ご注文ください。)

X68000シリーズと各種モニター組み合わせで大特価!!¥288,000より

OS-9/X68000
¥29,800

●X68000の高機能をフルにサポートしたリアルタイム・マルチタスク・オペレーティング・システム

リアルタイム・マルチタスク/メモリサイズの許す限り複数のタスクを起動でき、多数のI/Oデバイスがリアルタイム・マルチタスク機能でサポート。

マルチタスク・マルチウインドウ/オーバーラッピングタイプの本格的なマルチタスク・マルチウインドウがサポートされています。

AV-RIDER/OS-9のAV機能をフルに活用したサンプルソフトで、オーディオファイル、ビデオファイルをセレクト実行することができます。

マルチメディア/プレゼンテーション・サポート・システムによりADPCM(音声)、FM音源をグラフィックスと同時再生処理することが可能。

ファイル互換/Human68Kのリードライトをサポートしていますので、既にお持ちのデータを活用できます。

日本語処理/Human68Kと同様のASK68Kが使用できます。また、使い込んだ辞書ファイルもファイルコンバートすることにより、OS-9で利用出来ます。

●開発ツール「C&プロフェッショナル・パッケージ」¥58,000 同時発売(マイクロソフトウェアージャパン)

●熱転写カラー漢字プリンタ
シャープMZ-1P17
¥79,800⇒¥39,800
(ケーブル付き/第二水準漢字ROM ¥10,000別売)
※MZ/CZ全シリーズ対応、MZ80シリーズ不可。

●熱転写カラー80桁漢字プリンタ
シャープCZ-8PC2
¥69,800⇒¥48,000
(第二水準漢字ROM/ケーブル付き)
※X1シリーズ、X1turboシリーズ、X68000シリーズ使用可。

●24ピン80桁漢字ドットプリンタ
シャープCZ-8PK5
¥129,000⇒¥59,800
(第二水準漢字ROM/ケーブル付き)
※X1シリーズ、X1turboシリーズ、X68000

●24ピン136桁漢字ドットプリンタ
シャープCZ-8PK6
¥159,000⇒¥69,800
(第二水準漢字ROM/ケーブル付き)
※X1シリーズ、X!turboシリーズ、X68000シリーズ使用可。

アイビット推奨ディスプレイ

●富士通ゼネラルDM405
(14型)
(2000アナログ21/8ピン)
定価¥67,800⇒
特価¥36,000

DM405対応パソコン機種:MSX2。X1シリーズ。MZ700/1500/2000/2200シリーズ。FM77AV/7/8シリーズ。(ケーブルは各専用のものを使用)
MSX用21Pケーブルサービス!
(2/15~4/15)

●シャープCZ-880D-GY
(14型)TV付(2000/4000)
(デジタル/アナログ)
定価¥109,800⇒
特価¥69,800

CZ880DGY対応パソコン機種:CZ880C/881C。X1/TURBOシリーズ。ケーブルは本体付属を使用。PC88VA/VA2/VA3/MK2SR/TR/FR/MR。PC9801U/UV/UX/VM/VX/LV各シリーズ。アナログ25ピン↔25ピンケーブルを使用(デジタルは各専用ケーブルで)。MZ700/1500/2000/2200/2500各シリーズ(推奨品シャープ8D8K)。

●シャープCZ-820D
(14型)TV付
(2000デジタル)
定価¥79,800⇒
特価¥39,800

CZ820D対応パソコン機種:CZ880C/881C。X1/TURBOシリーズ(X1モードのみ)ケーブルは付属を使用。MZ700/1500/2000/2200シリーズ(推奨品シャープ8D8K)。その他デジタル表示は各専用ケーブルで。

●シャープCu-15M1
(15型デジタル/アナログ)
定価¥99,800⇒
特価¥79,800

Cu-15M1対応パソコン機種:CZ880C/881C。X1/TURBOシリーズ。ケーブルは本体付属を使用。PC88VA/VA2/VA3/MK2SR/TR/FR/MR。PC9801U/UV/UX/VM/VX/LV各シリーズ。アナログ25ピン↔25ピンケーブルを使用(デジタルは各専用ケーブルで)。MZ700/1500/2000/2200/2500各シリーズ(推奨品シャープ8D8K)。

●シャープCu14BD
(14型)(2000/4000)
(アナログ)
定価¥64,800⇒
特価¥49,800

Cu14BD対応パソコン機種:CZ880C/881C(X1/TURBOシリーズはAN506使用)。PC88/VA2/VA3/MkIISR/MR/FR/TR。PC8801FH/MH/FA/MA。PC286U/V/L。PC9801UV/UX/VM/VX/LV各シリーズは付属ケーブルを使用。

本 体

- ●シャープCZ-822C……¥59,800
- ●シャープ CZ-888 C-BK(X1 turbo ZIII) 新発売
- ●シャープCZ-880C……¥218,000⇒¥95,000
- ●シャープMZ-2861+1P-1252……¥383,000⇒¥245,000
- ●シャープMZ-2520……¥159,800⇒¥78,000
- ●シャープMZ-2521……¥198,000⇒¥85,000
- ●NEC PC-9801VX4……¥643,000⇒¥360,000
- ●NEC PC-9801XA2……¥695,000⇒¥149,000
- ●NEC PC-98LT11……¥238,000⇒¥119,800
- ●富士通FM-AV771……¥128,000⇒¥45,000
- ●富士通FM-AV772……¥158,000⇒¥55,000
- ●富士通AM-AV40……¥228,000⇒¥95,000
- ●富士通16βFD……¥400,000⇒¥180,000
- ●富士通16βキーボード……¥25,000⇒¥20,000

拡張機器他

- ●シャープCZ-8TM1……¥29,800⇒¥9,800
- ●シャープMZ-1E29……¥17,800⇒¥9,800
- ●シャープX1用ジョイカード……¥1,500
- ●シャープCZ-8EB3(I/Oボックス)……¥33,800⇒¥28,000
- ●シャープCZ-8EP(I/Oポート)……¥11,800⇒¥9,000
- ●シャープMZ-1U09……¥9,000⇒¥7,200
- ●シャープMZ-1E24232Cカード……¥19,800⇒¥16,800
- ●シャープCZ-8BK3……¥13,800⇒¥11,700
- ●シャープCZ-8BK4……¥6,800⇒¥5,700
- ●シャープMZ-1M03……¥69,000⇒¥35,000
- ●シャープMZ8BC04……¥18,000⇒¥8,000
- ●シャープMZ-8B104……¥45,000⇒¥18,000
- ●シャープMZ-1R09……¥35,000⇒¥25,000
- ●シャープMZ-1R10……¥30,000⇒¥12,000
- ●シャープMZ-1R11……¥80,000⇒¥40,000
- ●シャープMZ-1R19……¥35,000⇒¥15,000
- ●シャープMZ-1R24……¥22,000⇒¥6,000
- ●シャープMZ-1R26A……¥15,000⇒¥12,800
- ●シャープMZ-1R27A……¥13,000⇒¥10,000
- ●シャープMZ-1R28A……¥13,000⇒¥10,000
- ●シャープMZ-1R29A……¥32,000⇒¥10,000
- ●シャープMZ-1R37……¥35,800⇒¥28,000
- ●シャープMZ-1T02……¥19,800⇒¥8,500
- ●シャープMZ-1T03……¥12,000⇒¥8,500
- ●シャープCZ-8BGR2……¥14,800⇒¥4,000
- ●シャープCZ-8BS1……¥23,800⇒¥19,500
- ●シャープMZ-2000/2200/80B/1500/700用(フロッピーインターフェース)¥23,500⇒¥18,000
- ●シャープX1、MZ用マウス……特価¥4,800
- ●シャープMZ-1X29……¥13,800⇒¥11,000
- ●シャープMZ-1M08……¥10,000⇒¥6,000
- ●シャープMZ-3500キーボード……¥10,000
- ●シャープMZ-5500キーボード……¥10,000
- ●シャープX1シリーズ用キーボード……¥10,000
- ●シャープMZ-2000/2200通信セット MZ-1E29+MZ-1X22+MZ-2Z052……¥49,100⇒¥20,000
- ●シャープMZ-1E26……¥24,800⇒¥13,000

プリンター

- ●シャープMZ-1P27……¥268,000⇒¥214,400
- ●シャープMZ-1P28……¥148,000⇒¥118,400
- ●シャープMZ-1P29……¥168,000⇒¥134,400
- ●シャープMZ-1P17(ケーブルプリンター)¥85,800⇒¥39,800
- ●シャープ6P-11(カットシードヒート)……¥95,000⇒¥35,000

- ●シャープCZ-8PC2……¥69,800⇒¥49,800
- ●シャープCZ-8PC3……¥65,800⇒¥52,000
- ●シャープCZ-8PD2……¥79,800⇒¥25,000
- ●シャープMZ-8PD3……¥59,800⇒¥16,000
- ●シャープMZ-1P10A……¥245,000⇒¥80,000
- ●シャープCZ-8PK5……¥129,000⇒¥59,800
- ●シャープCZ-8PK6……¥159,000⇒¥69,800
- ●NEC-NM9700(漢字プリンタ)……¥163,000⇒¥88,000

ハードディスク

- ●シャープMZ-2500対応(20MB)(インターフェイス付き)……¥31,000
- ●シャープMZ-2800対応(20MB)(インターフェイス付き)……¥31,000
- ●シャープX68000対応ITX203(20MB)……¥125,000⇒¥95,000
- ●シャープX68000対応ITX403(40MB)……¥188,000⇒¥130,000

フロッピーディスク

- ●シャープCZ-503F……¥49,800⇒¥34,000
- ●シャープCZ-503F(インターフェースカードなし)……¥30,000
- ●シャープCZ-502F……¥99,800⇒¥75,000
- ●シャープCZ-300F(CZ-3PCM付)……¥13,000

ソフト

- ●シャープCZ-141SF(X1Z)……¥18,800⇒¥16,000
- ●シャープMZ-2Z013(MZ-5500)¥25,000⇒¥21,000
- ●シャープMZ-2Z017(MZ-5500)¥20,000⇒¥17,000
- ●シャープMZ-4Z001(5500/6500 IBM)・¥30,000⇒¥8,000
- ●シャープMZ-2Z064(MZ-6500)¥69,800⇒¥59,500
- ●シャープMZ-2Z023(MZ-5500)¥50,000⇒¥42,500
- ●シャープMZ-2Z025(MZ-5500)¥49,800⇒¥15,000
- ●シャープMZ-2Z014……¥68,000⇒¥15,000
- ●シャープMZ5Z013(MZ-5500)……¥6,500⇒¥2,000
- ●シャープ6F03(MZ-1500)……10枚¥4,000
- ●シャープMZ-6Z010(MZ-2500)・¥10,000⇒¥8,500
- ●シャープMZ-1M01(200/16ビット)……特価¥8,500

X68000関係ソフト

- ●CZ-220BS……¥58,000⇒ 大特価!
- ●CZ-226BS……¥29,800⇒ 大特価!
- ●シャープCZ-21SMS(サンプリングPRO68K)……¥17,800⇒ 大特価!
- ●CZ-227BS…… 大特価!
- ●シャープCZ-211LS……¥39,800⇒ 大特価!
- ●シャープCZ-6BE1……¥35,000⇒ 大特価!
- ●シャープCZ-6BE1A……¥38,000⇒ 大特価!

SHARPポケットコンピュータ

- ●PC-1360……¥29,800⇒¥19,800
- ●PCE-200……¥22,000⇒¥17,800
- ●PCE-500……¥28,800⇒¥24,800
- ●CE-152……¥19,800⇒¥9,800
- ●CE-161プログラムモジュール……¥50,000⇒2個¥7,000
- ●CE-159プログラムモジュール……¥35,000⇒¥4,200
- ●シャープ CE-140P……¥43,000⇒16,000

ポケコン総合カタログ並びに特価表を差し上げます。切手¥70を同封の上、当社へお申込みください。

本誌発売時には、上記価格よりさらにお求めやすい価格に変更されている場合があります。

☎0426-45-3001~3

FAX.0426-44-6002

●営業時間/10:00~19:00●電話受付/20:00迄可●定休日/日曜日(祭日営業)

SHARP SUPER XEX SHOP

アイビット電子株式会社 〒192 東京都八王子市北野町560-5

信用をモットーに、よりよい品をより安く、迅速にお届けします。

北海道から沖縄まで

★送料はご注文の際にお問い合わせ下さい。
★掲載の商品は、すべて新品、保証書付きです。
★掲載の商品は充分用意してありますが、ご注文の際は、在庫の確認の上、現金書留または、銀行振込でお申し込み下さい。全商品クレジットでも扱っております。
★お申し込みの際は必ず電話番号を明記して下さい。
★商品、品切れの節はご容赦下さい。

富士銀行八王子支店 (普)1752505

クレジット
金利大幅
ダウン!!

COMPUTER BANK

J-DMA 安心と信頼のシステムで新時代を切り開く

X68000

"ついにベールが剝された!"68000CPU搭載。ひとつひとつのスペックに新鮮な驚きがある。未体験の機能美が創造力を刺激する。

☆注文No.A-0321

SHARP CZ-601C	¥319,800
SHARP CZ-601D	¥119,800
標準価格合計	¥439,600
現金特別価格	~~¥439,600~~

大特価にて提供中

☆注文No.A-0322

SHARP CZ-611C	¥399,800
SHARP CZ-601D	¥119,800
標準価格合計	¥519,600
現金特別価格	~~¥519,600~~

大特価にて提供中

☆注文No.A-0323

SHARP CZ-601C	¥319,800
SHARP CZ-601D	¥119,800
SHARP CZ-6STI(チルトスタンド)	¥5,800
標準価格合計	¥445,400
現金特別価格	~~¥445,400~~

大特価にて提供中

☆注文No.A-0324

SHARP CZ-611C	¥399,800
SHARP CZ-601D	¥119,800
SHARP CZ-6STI(チルトスタンド)	¥5,800
SHARP CZ-6VTI(カラーイメージユニット)	¥69,800
標準価格合計	¥595,200
現金特別価格	~~¥595,200~~

大特価にて提供中

当社は X68000 PRO SHOPです。

■周辺機器　大特価にて提供中

品番	品名・内容	定価
CZ-601D	15型カラーディスプレイテレビ	¥119,800
CZ-611D	15型カラーディスプレイテレビ	¥145,000
CZ-603D	14型カラーディスプレイ	¥84,800
CZ-6STI	601D・611D用チルトスタンド	¥5,800
CU-21CD	21型カラーディスプレイ	¥139,800
CZ-6TU	RGBシステムチューナー	¥35,800
BF-68PRO	601・611・603用CRTフィルター	¥19,800
CZ-6VTI	カラーイメージユニット	¥69,800

型番	品名・内容	定価
CZ-8NSI	カラーイメージスキャナ	¥188,000
CZ-6BNI	スキャナ用パラレルボード	¥29,800
CZ-6BE1A	1MB増設RAMボード(内蔵用)	¥38,000
CZ-6BE2	2MB増設RAMボード(内蔵用)	¥79,800
CZ-6BE4	4MB増設RAMボード(内蔵用)	¥138,000
CZ-6BUI	ユバーサルI/Oボード	¥39,800
CZ-6BGI	GP-IBボード	¥59,800
CZ-6BFI	増設用RS-232Cボード(2ch)	¥49,800

型番	品名・内容	定価
CZ-6BPI	数値演算プロセッサボード	¥79,800
CZ-6BCI	FAXボード	¥79,800
CZ-6BMI	MIDIボード	¥26,800
CZ-6EBI	拡張I/Oボックス(4スロット)	¥88,000
CZ-6PVI	カラービデオプリンタ	¥198,000
CZ-6BUI	ユバーサルI/Oボード	¥39,800
CZ-620H	ハードディスクユニット(20MB)	¥178,000
AN-160SP	アンプ内蔵スピーカーシステム(2本1組)	¥59,800

■ソフトウェア　大特価にて提供中

メーカー名	型番	品名・内容	定価
SHARP	CZ-212BS	BUSINESS PRO-68K	¥68,000
SHARP	CZ-220BS	DATA PRO-68K	¥58,000
SHARP	CZ-226BS	CARD PRO-68K	¥29,800
SHARP	CZ-214MS	SOUND PRO-68K	¥15,800
SHARP	CZ-213MS	MUSIC PRO-68K	¥18,800
SHARP	CZ-215MS	Sampling PRO-68K	¥17,800

メーカー名	型番	品名・内容	定価
SHARP	CZ-237MS	Musicstudio PRO-68K	¥25,000
SHARP	CZ-247MS	MUSIC PRO-68K(MIDI)	近日発売
SHARP	CZ-221HS	NEW Print Shop PRO-68K	¥19,800
SHARP	CZ-223CS	Communication PRO-68K	¥19,800
SHARP	CZ-211LS	C compiler PRO-68K	¥39,800
SHARP	CZ-231AS	フルスロットル	¥8,800

メーカー名	型名	品名・内容	定価
SHARP	CZ-232AS	熱血高校ドッジボール部	¥7,800
SHARP	CZ-218AS	沙羅曼蛇	¥8,800
電波新聞社		ドラゴンスピリット	¥8,800
テクノソフト		サンダーフォースII	¥9,800
ツァイト		Z'sSTAFF PRO-68K	¥58,000

●どこよりもお得な高額下取り実施中!!　セットの組合わせは自由自在、ぜひご相談下さい。

X1 turbo Z III

画像取り込み、ビデオ編集、ステレオFM音源、多才な機能でひろがるアートワーク。

☆注文No.A-0325

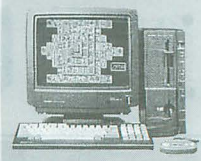
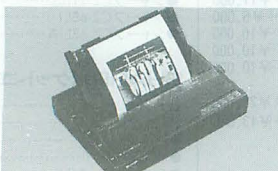
SHARP CZ-888C-BK	¥169,800
SHARP CZ-860D-BK	¥99,800
標準価格合計	¥269,600
現金特別価格	~~¥269,600~~

大特価にて提供中

X1 twin

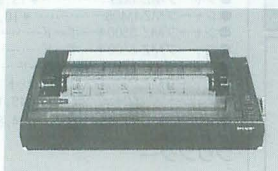
HEシステム(PC Engine)搭載で楽しさ2倍

☆注文No.A-0326

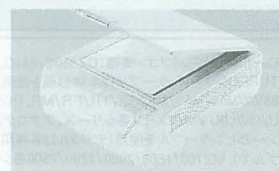
SHARP CZ-830CBK	¥99,800
SHARP CZ-820DB	¥79,800
標準価格合計	¥179,600
現金特別価格	¥109,800

●どこよりもお得な高額下取り実施中!!　セットの組合わせは自由自在、ぜひご相談下さい。

☆注文No.B-0323

SHARP CZ-8PC3	¥65,800
現金特別価格	~~¥65,800~~

大特価にて提供中

■お支払例
①¥9,700×6回〔ボーナス〕無し
②¥3,100×20回〔ボーナス〕無し

☆注文No.B-0324

SHARP CZ-8PK6	¥159,000
現金特別価格	¥59,800

■お支払例
①¥6,300×10回〔ボーナス〕無し
②¥3,300×24回〔ボーナス〕無し

☆注文No.B-0325

SHARP JX-200	¥198,000
現金特別価格	~~¥198,000~~

大特価にて提供中

■お支払例
①¥9,400×20回〔ボーナス〕無し
②¥5,600×36回〔ボーナス〕無し

☆注文No.B-1132

SHARP AN-8TU	¥35,800
現金特別価格	~~¥35,800~~

大特価にて提供中

●どんな問い合わせにも親切に対応いたします。

全商品保証付 中古も6ヶ月の保証期間だから安心です。

クレジットでOK カレッジクレジットも取扱います。

全国無料配送 お買上1万円以上、配達料はいただきません。

日曜配達可 留守の多い方でも安心です。

ショールーム Xシリーズ展示中。

高額買取り 電話1本で即、現金お支払い。

代金引換えシステム 商品到着時の代金支払いでOK。

ボーナス一括払い 商品は即お手元へ、お支払いはボーナス時に。

話題の新製品が全国どこでも電話で買える!!

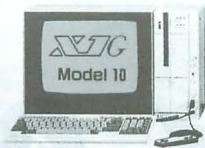
03(797)1221

Let's CALL

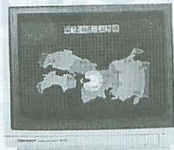
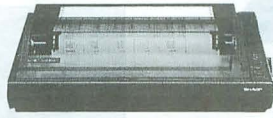
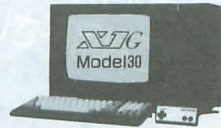
超優良中古パソコンが電話一本で買える!!

03(797)1221

SHARP
CZ-880DGY 新品同様
(14インチ400/200RGBTV)
¥109,800➡¥69,800
(色はグレーになります。)

SHARP
CZ-880CB (X-1TurboZ本体)
¥218,000➡¥74,800
CZ-880DB 新品同様
¥109,800➡¥85,000
セット価格 ¥327,800➡¥159,800

SHARP
CZ-820CE
(X-1Gモデル10) 新品同様
¥69,800➡¥9,800
X-1Gモデル10RFコンバータセット
(本体+AN-58C) 新品同様
¥72,780➡¥12,600
X-1GモデルTVディスプレイセット
(本体+CZ-820D) 新品同様
¥119,600➡¥39,800

SHARP
CU-14ED 新品
(14インチ4050/2000字RGB、PC用アナログRGBケーブル付)
¥79,800➡¥49,800

SHARP
CZ-820DE・B 新品
(14インチ2000字RGBTV)
¥79,800➡¥39,800

SHARP
CU-14GB/E 新品
(14インチ2000字デジタルRGB)
¥49,800➡¥29,800

SHARP
CZ-8PK6 新品同様
(15インチ漢字プリンタ)
¥159,000➡¥59,800

SHARP
CZ-822C
(X-1Gモデル30本体) 新品同様
¥118,000➡¥49,800
X-1Gモデル30TVディスプレイセット
(本体+CZ-820D) 特選極上品
¥197,800➡¥79,800

SHARP

本体

品名	定価	販売価格
CZ-812C(X-1F model 20)	¥139,800	¥32,000
CZ-822C(X-1G model 30)	¥118,000	¥42,000
CZ-830C(X-1Twin)	¥99,800	¥52,000
CZ-850CB(X-1Turbo モデル10)	¥168,000	¥22,000
CZ-880CB(X-1Turbo Z)	¥218,000	¥74,800
MZ-2521	¥198,000	¥48,000
MZ-2531	¥198,000	¥62,000

ディスプレイ

品名	定価	販売価格
12M-314C(14"カラー4050文字)	¥128,000	¥45,000
14M-111C(14"カラー1000文字)	¥67,800	¥15,000
CU-14H2(14"カラー4050文字)	¥99,800	¥45,000
CZ-855D(14"カラー4050/2000文字RGBTV)	¥119,800	¥58,000
CZ-880D-GY(14"カラー4050/2000文字RGBTV)	¥109,800	¥64,800
MD-12P1(12"グリーン4050文字)	¥39,800	¥22,000
MZ-1D22(14"カラー4050文字)	¥108,000	¥45,000

ディスクドライブ・プリンタ・他

品名	定価	販売価格
CZ-801F(5"2D、2ドライブ)	¥198,000	¥35,000
CZ-81P(ミニサイズプリンタ)	¥34,800	¥10,000
CZ-8PP2(カラープロッタプリンタ) 新品	¥54,800	¥15,000
CZ-8PD2(10"ドットプリンタ)	¥79,800	¥28,000
CZ-8PD3(10"ドットプリンタ)	¥59,800	¥28,000
CZ-8PK6(15"24ドット漢字プリンタ) 新品	¥159,000	¥59,800
MZ-1P06(80桁漢字プリンタ)	¥234,000	¥45,000
MZ-1P17(80桁24ドットカラー漢字熱転写プリンタ)	¥79,800	¥30,000
CZ-8SS2(システムスタンド)	¥5,500	¥4,000
CZ-8BS1(FM音源ボード) 新品	¥23,800	¥20,000
CZ-8NM2(マウス)	¥6,800	¥4,000

＊SHARP X-1シリーズ特選極上品コーナー＊

品名	定価	販売価格
CZ-820CE(X-1G/10) 新品同様	¥69,800	¥9,800
CZ-822CB(X-1G/30) 新品同様	¥118,000	¥49,800
CZ-822C+CZ-820D(X-1G/30セット) 特選極上品	¥197,800	¥79,800

＊SHARP ディスプレイ特選極上品コーナー＊

品名	定価	販売価格
CU-14G(14"カラー2000文字) 新品	¥49,800	¥29,800
CU-14ED(14"カラー4050/2000文字) 新品	¥79,800	¥49,800
CZ-820D(14"カラー2000文字RGBTV) 新品同様	¥79,800	¥39,800
CZ-880DB(15"カラー4050文字RGBTV) 新品同様	¥109,800	¥85,000
CZ-880DGY(15"カラー4050文字RGBTV) 新品同様	¥109,800	¥69,800

6つの安心のアフターサービス

1 C.B.クラブ

■あなたも今すぐ会員に!!

当社で商品をお買い上げの方全員に、C.B.クラブカードを無料でお送り致します。このカードをお持ちの方なら次の買い換え時や、付属品の購入時に会員特別価格でご購入になれます。

2 C.B.サポートホットライン

☎03(797)1234

■トラブルへの対応!!

当社でコンピュータをお買い上げいただいたお客様に万一、トラブルが発生した場合、このホットラインで親切に対応致します。

3 C.B.レスキューシステム

■迅速なサポート体制!!

お客様のお手元でトラブルが発生した場合、当社より引取りにお伺い致します。万一、お買いになった機械が故障しても安心です。

4 C.B.クイック・チェンジシステム

■新品交換体制も万全!!

お買い上げになったパソコンが、万一初期不良でも安心です。商品到着後7日以内にご連絡いただければ、新品と交換致します。

5 RX2アフターサポート

■PC-9801愛好家にお得です!!

NEC RX2をお買い上げいただいたお客様に保証期間中、万一故障があった場合無料で代品を貸出します。

6 C.B.Q&Aホットライン

☎03(797)1233

■素朴な疑問何でもどうぞ!!

ハードウェア、ソフトウェアに関するご質問なら内容を問わずどなたからでも親切に、ご相談をお受け致しております。

- 電話一本で高額下取り、即商品はお手元へ!
- あなたの不要になったパソコンを電話一本で査定し買取ります。
- 掲載の商品以外も取り扱っております。
- ビジネスソフトスクール受講者受付中!お気軽にお電話下さい。

▼本社注文デスク
03(797)1221

コンピュータバンク

株式会社パシフィックコンピュータバンク 〒150 東京都渋谷区渋谷1-6-8 井上ビル 営業時間/AM9:30~PM9:30 年中無休

超低金利クレジットOK.!

うれしい!!
夏のボーナス一括払い
手数料(金利)無料
(7月支払、ご利用下さい)

またまた秋葉原でおなじみの

2/15〜3/20

●お近くの方はお
●本体単品で特
●ビジネスソフト定

特　報.!!
中古パソコンの現金買取り
下取りOK.!!
(差額は低金利クレジットをご利用下さい。)

X68000ACE HD　(送料¥2,000)

NEW
CZ-603D
(定価¥84,800)
●0.31ピッチ
●14インチ
●TVチューナーなし

Ⓐセット:CZ-611C+CZ-611D+M-2HD(10枚)+ゲーム
………定価¥544,800➡P&A超特価(価格はお電話下さい)

12回	?	24回	?	36回	?	48回	?	60回	?

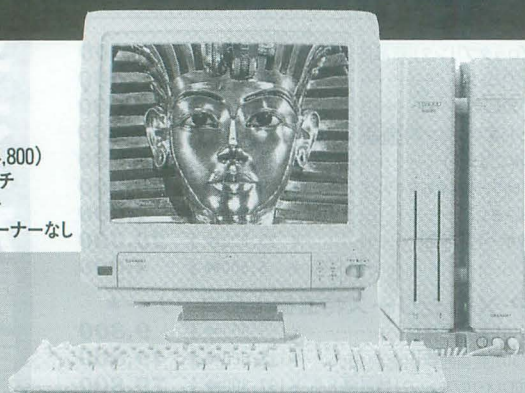
Ⓑセット:CZ-611C+CZ-601D+M-2HD(10枚)+ゲーム
………定価¥519,600➡P&A超特価(価格はお電話下さい)

12回	?	24回	?	36回	?	48回	?	60回	?

Ⓒセット:CZ-611C+CZ-603D+M-2HD(10枚)+ゲーム
………定価¥484,600➡P&A超特価

Ⓓセット:CZ-611C+Cu-21CD+M-2HD(10枚)+ゲーム
………定価¥539,600➡P&A超特価(価格はお電話下さい)

※X68000セットでお買い上げの方にゲームの他にドラゴンスピリッツ(¥8,800)をプレゼント!
※チルトスタンド(CZ6ST1 ¥5,800)必要な方は¥5,000加算して下さい。

X68000ACE　(送料¥2,000)

X68000用
ジョイスティック
送料¥500
●XE-1PRO
定価¥9500
特価¥7,800
●ASCII STICK
X-TURBO
定価¥6,800▶
特価¥5,500

Ⓐセット:CZ-601C+CZ-611D+M-2HD(10枚)
………定価¥464,800➡P&A超特価(価格はお電話下さい)

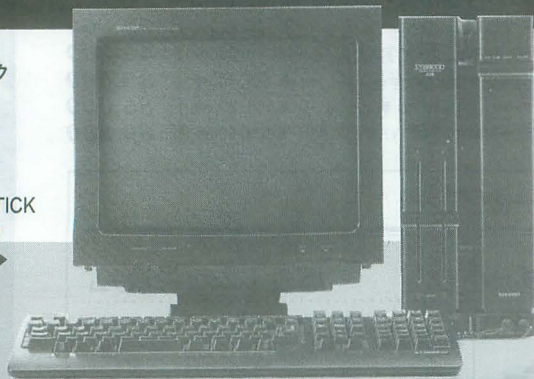
12回	?	24回	?	36回	?	48回	?	60回	?

Ⓑセット:CZ-601C+601D+M-2HD(10枚)
………定価¥439,600➡P&A超特価(価格はお電話下さい)

12回	?	24回	?	36回	?	48回	?	60回	?

Ⓒセット:CZ-601C+CZ-603D+M-2HD(10枚)
………定価¥404,600➡P&A超特価(価格はお電話下さい)

Ⓓセット:CZ-601C+Cu-21CD+M-2HD(10枚)+ゲーム
………定価¥459,600➡P&A超特価(価格はお電話下さい)

※チルトスタンド(CZ-6STI ¥5,800)必要な方は¥5,000加算して下さい。
※X68000セットでお買い上げの方にゲームの他にドラゴンスピリッツ(¥8,800)をプレゼント!

X-1ターボZⅢ/ZⅡ/Z (セットでお買い上げの方にディスケット10枚 ジョイカード、ゲーム3種プレゼント) 送料¥2,000

Ⓐセット:NEW
X-1ターボZⅢ(CZ-888C+CZ-860D)
定価¥269,600▶価格はお電話下さい

12回	?	24回	?	36回	?	48回	?	60回	?

Ⓑセット:NEW
X-1ターボZⅢ(CZ-888C+CZ-830D)
定価¥269,600▶価格はお電話下さい

12回	?	24回	?	36回	?	48回	?	60回	?

Ⓒセット:X-1ターボZⅡ
(CZ-881C+CZ-880D)
定価¥289,600▶特価**¥182,000**

Ⓓセット:X-1ターボZ
(CZ-880C+CZ-880D)
定価¥327,800▶特価**¥158,000**

X-1 TWIN/G　(送料¥2,000)

Ⓐセット:X-1 TWIN(CZ-830C+CZ-820D)…………定価¥179,600▶特価**¥94,000**
Ⓑセット:X-1Gモデル30(CZ-822C+CZ-820D)……定価¥197,800▶特価**¥79,000**
●セットでお買い上げの方に、ディスケット10枚、ジョイカード、ゲーム3種、パソコンラックⒶ3段をプレゼント

プリンターセット ※全セットにケーブル、用紙付　(送料¥1,000)

Ⓐセット:CZ-8PC2 限定…………………定価¥69,800▶特価**¥44,000**
Ⓑセット:CZ-8PC3…………………………定価¥65,800▶P&A超特価(お電話下さい)
Ⓒセット:CZ-8PK7…………定価¥122,000▶P&A超特価(お電話下さい)

12回	?	24回	?	36回	?

Ⓓセット:CZ-8PK8…………定価¥152,000▶P&A超特価(お電話下さい)

12回	?	24回	?	36回	?

Ⓔセット:CZ-8PK9…………定価¥89,800▶P&A超特価(お電話下さい)

12回	?	24回	?

Ⓕセット:CZ-8PK6…………………定価¥159,000▶超特価**¥69,000**
限定品、用紙1,000枚付、送料無料

1回～60回払いまでOK!!

★頭金なし! ★即日発送

P&AがズバリスーパーKA超特価セールでご奉仕!!

立寄り下さい。専門係員が説明いたします。
価で受付します。詳しくは電話にてお問合せ下さい。
価の20%引きOK! TELください。

全国通販

X68000用ソフトコーナー(送料￥1,000)

Ⓐ CZ-212BS(BUSINESS)……定価￥ 68,000➡特価￥55,000
Ⓑ CZ-220SB(DATA)……定価￥ 58,000➡P&A超特価
Ⓒ CZ-226BS(CARD)……定価￥ 29,800➡P&A超特価
Ⓓ CZ-213MS(MUSIC)……定価￥ 18,800➡特価￥15,000
Ⓔ CZ-214MS(SOUND)……定価￥ 15,800➡特価￥12,500
Ⓕ CZ-215MS(Sampling)……定価￥ 17,800➡特価￥14,000
Ⓖ CZ-221HS(NEW Print shop)……定価￥ 19,800➡P&A超特価
Ⓗ CZ-223CS(Communication)……定価￥ 19,800➡P&A超特価
Ⓘ CZ-211LS(C. compiler)……定価￥ 39,800➡特価￥32,000
Ⓙ CZ-224LS(福袋)……定価￥ 9,800➡特価￥ 8,000
Ⓚ Z's STAFF PRO-68K(シャフト)……定価￥ 58,000➡特価￥42,000
Ⓛ 神風(サムシンググッド)……定価￥ 68,000➡特価￥49,000
Ⓜ ビジネスAD68K(マッショシステム)……定価￥ 98,000➡特価￥78,500
Ⓝ 弥生(日本マイコン)……定価￥ 80,000➡特価￥64,000
Ⓞ CP/M-68K(ニューウェイブ)……定価￥110,000➡特価￥88,000
Ⓟ EW&EI(イースト)……定価￥ 38,000➡特価￥30,500
Ⓠ C-TRACE(キャスト)……定価￥ 68,000➡特価￥54,500
Ⓡ SHOGUN(サムシンググッド)……定価￥ 34,800➡特価￥25,000
Ⓢ SAMURAI(サムシンググッド)……定価￥ 19,800➡特価￥15,200

カラービデオプリンター (送料￥1,000)

Ⓐセット:CZ-6PVI……定価￥198,000➡超特価￥155,000

12回	13,400	24回	7,000	36回	4,800	48回	3,700

カラーイメージスキャナ (送料￥1,000)

Ⓐセット:CZ-8NSI……定価￥188,000➡超特価￥145,000

12回	12,600	24回	6,600	36回	4,500	48回	3,500

周辺機器コーナー(送料￥1,000)●その他の周辺機器はお電話下さい。

Ⓐ CZ-8BSI(FM音源ボード)……定価￥23,800➡特価￥19,000
Ⓑ CZ-8RLI(データレコーダ)……定価￥24,800➡特価￥20,000
Ⓒ CZ-8AV2(カラーイメージボードⅡ)……定価￥39,800➡特価￥31,000
Ⓓ CZ-8BRI(立体映像セット)……定価￥29,800➡特価￥23,000
Ⓔ CZ-8DT2(パーソナルテロッパ)……定価￥44,800➡特価￥35,000
Ⓕ CZ-6VTI(カラーイメージユニット)……定価￥69,800➡P&A超特価
Ⓖ CZ-6EBI(拡張I/Oボックス)……定価￥88,000➡特価￥69,000
Ⓗ AN-160SP(アンプ内蔵スピーカーシステム)……定価￥59,800➡特価￥47,000

ゲームソフト(1ヶ～20ヶまで送料￥500)

X68000用
Ⓐ 源平討魔伝(電波新聞社)……定価￥ 7,800➡特価￥ 6,200
Ⓑ ドラゴンスピリット(電波新聞社)定価￥ 8,800➡特価￥ 7,000
Ⓒ スペースハリアー(電波新聞社)定価￥ 6,800➡特価￥ 5,400
Ⓓ 熱血高校ドッジボール部(SHARP)……定価￥ 7,800➡P&A超特価
Ⓔ 沙羅曼蛇(SHARP)……定価￥ 8,800➡P&A超特価
Ⓕ フルスロットル(SHARP)……定価￥ 8,800➡P&A超特価
Ⓖ 琥珀色の遺言(リバーヒルソフト)・定価￥ 9,800➡特価￥ 7,800
Ⓗ ザ・スーパーラスベガス(日本デグスタ)……定価￥12,800➡特価￥10,200
Ⓘ マイト・アンド・マジック(スタークラフト)定価￥ 9,800➡特価￥ 7,800
Ⓙ ザ・リターン・オブ・イシター(SPS)定価￥ 7,800➡特価￥ 6,200
Ⓚ 信長の野望(全国版)(KOEI)……定価￥ 9,800➡特価￥ 7,800
Ⓛ 麻雀悟空(シャノアール)……定価￥ 7,800➡特価￥ 6,200
Ⓜ マーダークラブDX(リバーヒルソフト)定価￥ 7,800➡特価￥ 6,200
Ⓝ ザキングオブシカゴ(ボーステック)……定価￥12,800➡特価￥10,200
Ⓞ 今夜も朝までパワフルまあじゃん2(dB-SOFT)……定価￥ 7,800➡特価￥ 6,200
Ⓟ 三国志(光栄)……定価￥14,800➡特価￥12,000

X68000用ビジネスソフト

Ⓐ Final Ver. 3.2(エー・エス・ピー)……定価￥38,000➡特価￥30,000
Ⓑ 彩 CRONE……定価￥58,000➡特価￥46,000
Ⓒ OS-9/X68000(シャープ)……定価￥29,800➡特価￥23,500

P & A 特選パソコンラック (送料無料)移動自由(キャスター付)

Ⓐ 3段
875(H)
×580(D)
×610(W)
￥9,000

Ⓑ 4段
￥12,000

Ⓒ 5段
1280(H)
×600(D)
×620(W)
￥16,500

通信ソフト&FAXアダプター

Ⓐ FF-P4800(アイワ)……定価￥98,000▶特価￥75,000
Ⓑ FR-1000(エプソン)……定価￥59,800▶特価￥46,000
Ⓒ CRC-FAX(C-リサーチ)……定価￥62,000▶特価￥48,000
Ⓓ まいとーく……定価￥28,000▶特価￥21,500
Ⓔ ベルサルク……定価￥19,800▶特価￥15,000
Ⓕ パーティ……定価￥24,000▶特価￥18,000

中古パソコンはP&Aにおまかせ!!

その場で高価現金買取り・高価下取りOK!!

■まずはお電話下さい。
03-651-1884
FAX:03-651-0141

■下取り・買取りでお急ぎの方、直接当社に来店、または、宅急便にてお送り下さい。

●下取りの場合……価格は常に変動していますので査定額をお電話で確認して下さい。(差額は、P&A超低金利クレジットをご利用下さい。)

●買取りの場合……現品が着き次第、2日以内に買取り金額を連絡し、振込み、又は書留でお送り致します。

●近郊の方は、P&A本店まで、直接お持ち下さい。
即金にて、￥1,000,000までお支払い致します。

通信販売お申し込みのご案内

〔現金一括でお申し込みの方〕
●商品名およびお客様の住所・氏名・電話番号をご記入の上、代金を当社まで、現金書留でお送りください。(プリンター・フロッピーの場合、本体使用機種名を明記のこと)

〔銀行振込でお申し込みの方〕
●銀行振込ご希望の方は必ずお振込みの前にお電話にてお客様のご住所・お名前・商品名等をお知らせください。
(電信扱いでお振込み下さい。)

〔振込先〕住友銀行　新小岩支店
当No.263914 ㈱ピー・アンド・エー

〔クレジットでお申し込みの方〕
●電話にてお申し込みください。クレジット申し込み用紙をお送りいたしますので、ご記入の上、当社までお送りください。
●現金特別価格でクレジットが利用できます。残金のみに金利がかかります。
●1回～60回払いまで出来ます。但し、1回のお支払い額は3,000円以上。

アフターサービス万全
全商品保証付。専門の担当者がお客様の立場で対応します。
初期不良、輸送トラブルetc.
万が一初期不良、輸送トラブルが発生しました際には、即交換させていただきます。

超低金利クレジット率

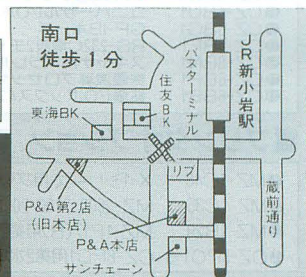
回数	1	3	6	10	12	15	18	24	36	48	60
利率(%)	1.5	2.0	3.0	4.5	4.5	7.5	9.0	9.5	13	17	22

●定休日/毎週水曜日＝第3水曜・木曜は連休とさせていただきます(祭日の場合は翌日になります)

南口
徒歩1分
JR新小岩駅
パスターミナル
住友BK
東海BK
リブ
蔵前通り
P&A第2店
(旧本店)
P&A本店
サンチェーン

●マイコン
●ビデオ
●ビデオテープ

株式会社ピー・アンド・エー
〒124 東京都葛飾区新小岩2丁目1番地19号
☎03-651-0148(代) FAX.03-651-0141

●営業時間 AM11:00～PM9:00
日・祭日も受付けます
(但しPM8:00迄)

パソコン通信

J&P HOT LINE でもお申し込みいただけます。

J&Pメールショッ

■あれば便利なグッズ

X3-1
東京ニーズ ベータ88MKII
ジョイスティック
J&P特価**10,000円**
PC-88用インターフェイスとジョイスティックセット

X3-2
クリーニングディスク
J&P価格**1,500円**
①5インチ ②3.5インチ

X3-3
CRTフィルター
メーカー標準価格19,000円
J&P特価 **15,000円**
HOYA アイテックフィルターエース14
14インチ用、静電気カット、フード付

X3-4
エレコムSO-450
J&P特価**3,300円**
原稿が見やすく場所をとりません

X3-5
（プリンタ別売）
プリンタスタンド
PS-10 ①10インチ用**2,000円**
PS-15 ②15インチ用**2,000円**

X3-6
MS-500
J&P特価**3,500円**
ディスプレイの角度を自由に調整できます。

X3-7
フロッピィケース
3.5インチ80枚収納可
J&P特価**2,000円**

X3-8
5インチ100枚収納可
J&P特価**2,000円**

X3-9
マウステーブル
J&P特価
キーボードの上でマウスが使え、伝票台にもできます。アーベル

X3-10
サンワ
電磁波防止エプロン
J&P価格**7,800円**

■パソコングッズ

X3-11
OA電源タップ
ナショナルWCH 4411
集中スイッチ付
J&P特価 **3,300円**

X3-12
エレコムSO-450
J&P特価**3,300円**
原稿が見やすく場所をとりません。

X3-13
5インチケース
100枚収納可
J&P特価**2,000円**

X3-14
3.5インチケース
80枚収納可
J&P特価**2,000円**

X3-15
プリンタスタンド
①10インチ用**2,300円**
②15インチ用**2,500円**

■ポケコン

X3-16
PC-E200
J&P特価**17,800円**
Z80CPU採用で高速演算を実現。24桁4行表示

X3-17
PC-E500
J&P特価**24,800円**
充実の124関数機能、最大96Kバイトまで増設可能。40桁4行表示

さあ始めようパソコン通信

■X-1通信セット

X3-18
モデム：CZ-8TM2
J&P HOTLINE:
スタータキット
通信速度300・1200bps
標準価格合計52,800円
セット価格**49,800円**

■X-1ターボ通信セット

X3-19
モデム：アイワ PV-A1200MKII
通信ソフト：SPS JETターボターミナル
J&P HOTLINE：スタータキット 通信速度300・1200bps
標準価格合計39,600円 セット価格**31,000円**

X3-20
シャープ
PA-8500
J&P特価 **24,800円**
これ1台で、電卓・電話帳・スケジュール管理・カレンダー・メモ・時計・世界時計・日数計算機能付。別売ICカードによりさらに使い道が広がります。

X3-22
シャープ
CE-60P
J&P特価 **22,800円**
電子手帳で入力した住所録・メモ・スケジュールのデータをハガキやノートに印字できます。
対応機種：PA8500／7000
6500／6000

X3-21
ICカード（PA-7000/8500共通）

❶PA-7C1	英和・和英カード	6,300円
❷PA-7C2	漢字辞書カード	9,000円
❸PA-7C3	6ケ国語会話カード	6,300円
❹PA-7C4	カラオケ歌詞カード	9,000円
❺PA-7C5	占い四柱推命カード	6,300円
❻PA-7C6	7ケ国語会話カード	6,300円
❼PA-7C8	シティガイド東京編	6,300円
❽PA-7C10	電話帳・住所録カード	9,000円
❾PA-7C11	販者管理カード	9,000円
❿PA-7C12	技術計算カード	6,300円
⓫PA-7C40	英和辞書カード	14,400円
⓬PA-7C41	国語辞典カード	14,400円

■〈X-1/ターボオプション〉

X3-23
マウス
シャープCZ-8NM2A
J&P価格 **6,800円**
X-1・MZ用マウス

X3-24
シャープCZ-8BV2
J&P価格**39,800円**
画像を自在に修正・加工できます
画像処理ツール・グラフィックソフト同梱

■ディスケット

マクセル
X3-27

❶MD2-D(10枚)	1,600円
❷MD2-DD(10枚)	1,700円
❸MD2-256HD(10枚)	1,800円
❹MF2-D(10枚)	3,400円
❺MF2-DD(10枚)	3,500円
❻MF2-256HD(10枚)	5,500円

X3-28
データライフディスケット
CMF-2DD(10枚)
2,700円

X3-29
データライフ
M-2HD256(10枚)
1,600円

X3-30
SONY
MF-2HD256(10枚)
4,500円

■X68000オプション X3-25

❶CZ-6BC1	FAXボード	79,800円
❷CZ-6BE1A	1MB増設メモリ(601C・611C)	38,000円
❸CZ-6BE1	1MB増設メモリ(600C)	35,000円
❹CZ-6BE2	2MB増設メモリ	79,800円
❺CZ-6BE4	4MB増設メモリ	138,000円
❻CZ-6BU1	ユニバーサルI/Oボード	39,800円
❼CZ-6BG1	GP-IBボード	59,800円
❽CZ-6BF1	RS-232C増設2チャンネル	49,800円
❾CZ-6BN1	スキャナ用パラレルボード	29,800円
❿CZ-6BP1	数値演算プロセッサボード	79,800円
⓫CZ-6BB1	拡張I/Oボックス4スロット	88,000円

■プリンタオプション X3-26

❶MZ-1C48	X-1シリーズ用プリンタケーブル	6,800円
❷MZ-1C35	MZ-2500/2200/2000用ケーブル	6,800円
❸MZ-1R29	MZ-1P17(B)用第2水準ROM	14,800円
❹CZ-8PC1-3	CZ-8PC1用第2水準ROM	9,800円

■ハンディコピー写楽 X3-31

ゼロックス

54,800円

104mm幅が人気／50・75・100・200%の倍率コピー可。12色の多色リボンが大好評。アクセサリーも充実し、ハンディコピーNo.1の実績です。

〈本体カラー〉
❶ブラック
❷ホワイト
❸ブルー

〈オプション〉
❹S309 ACパワーパック(ブラック) 9,800円
❺S310 ACパワーパック(ホワイト) 9,800円
❻S311 ACパワーパック(ブルー) 9,800円
❼S332 直線ガイド 4,000円
❽S334 ソフトケース 5,000円

●リボン
❾S315 12色セット 8,400円
❿S316 BK、R、B、G、Y、S 4,500円
⓫S317 BK、GLD、SIL・W・P・GY 4,500円
⓬S318 800円
⓭S319 800円
⓮S320 800円
⓯S321 グリーン 800円
⓰S322 イエロー 800円
⓱S323 セピア 800円
⓲S324 ゴールド 800円
⓳S325 シルバー 800円
⓴S326 ホワイト 800円
㉑S327 ピンク 800円
㉒S328 グレー 800円
㉓S329 ライトブルー 800円
㉔S330 透明3色セット 2,400円
㉕S331 ビビッド3色セット 2,400円

全国無料配達

ピング

メールショッピングのお申し込みは J&P 渋谷店で承ります。

東京都渋谷区道玄坂2丁目28番4号(〒150)
☎(03)496-4141〈水曜定休〉

■ホビーソフト

ドーム

注文No	X3-32
適応機種	X68000
ソフトハウス	システムサコム

文章データ20万字に秘められたシステムサコム自信の超新星ドームに描かれた反核二元論は人類存続への希望かもしれない。

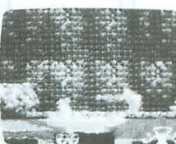
¥9,800(5"2HD)

クレイズ

注文No	X3-33
適応機種	X-1ターボ
ソフトハウス	ハート電子

核戦争の為、地下世界ににげ込んだ人間達。その中で巨大なコンピュータに支配される世界がつくり上げられた。ここでスーパーバイクをあやつる1人の男がいた。その名は"CRAZF"。3Dグラフィックの驚異の世界。

¥7,800(5"2D)

レジェンド

注文No	X3-34
適応機種	X-1シリーズ
ソフトハウス	クエイザーソフト

人の心の光と闇を司るクリスタルを妖精アリーナが誤って地上に落してしまった。そのクリスタルを手に入れたのは古しえの時代に神々をも滅ぼそうとした大魔王ガウディアであった。

¥7,800(5"2D)

蒼き狼と白き牝鹿ジンギスカン

注文No	X3-35
適応機種	MZ-2500
ソフトハウス	光栄

「蒼き狼と白き牝鹿」の壮大なストーリーに加え、戦闘モードでは騎馬隊や弓矢隊など新しく加えられた戦闘部隊や略奪、狩猟、降伏勧告などの新コマンドも加わって、より複雑な戦略が楽しめるシミュレーションゲームとして期待できる。

¥9,800(3.5"2DD)

注文No	タイトル	ソフトハウス	適応機種	メディア	価格
X3-36	サンダーフォースII	T&Eソフト	X68000	5"2HD	¥ 9,800
X3-37	信長の野望全国版	光栄	X68000	5"2HD	¥ 9,800
X3-38	マイト&マジック	スタークラフト	X68000	5"2HD	¥ 9,800
X3-39	サラマンダー	シャープ	X68000	5"2HD	¥ 7,800
X3-40	ドラゴンスピリット	電波新聞社	X68000	5"2HD	¥ 8,800
X3-41	琥珀色の遺言	リバーヒルソフト	X68000	5"2HD	¥ 9,800
X3-42	熱血高校ドッジボール	シャープ	X68000	5"2HD	¥ 7,800
X3-43	たんば	マイクロネット	X68000	5"2HD	¥ 7,800
X3-44	道化師殺人事件	シンキングラビット	X68000	5"2HD	¥ 7,800
X3-45	名監督II	コムパック	X68000	5"2HD	¥ 9,800
X3-46	上海	システムソフト	X68000	5"2HD	¥ 6,500
X3-47	ドーム	システムサコム	X68000	5"2HD	¥ 9,800
X3-48	源平討魔伝	電波新聞	X68000	5"2HD	¥ 7,800
X3-49	スペースハリアー	電波新聞	X68000	5"2HD	¥ 6,800
X3-50	マンハッタンレクイエム	リバーヒルソフト	X68000	5"2HD	¥ 7,800
X3-51	殺意の接吻	リバーヒルソフト	X68000	5"2HD	¥ 5,800
X3-52	ソフトでハードな物語	システムサコム	X68000	5"2HD	¥ 9,800
X3-53	リターンオブインター	SPS	X68000	5"2HD	¥ 7,800
X3-54	麻雀悟空	アスキー	X68000	5"2HD	¥ 7,800
X3-55	A列車で行こうII	アートディンク	X68000	5"2HD	¥12,800
X3-56	サイバライターVOL2	日本コンピュータ連盟	X68000	5"2HD	¥ 5,980
X3-57	花札放浪記	ドット企画	X68000	5"2HD	¥ 6,800
X3-58	アルカノイド	シャープ	X68000	5"2HD	¥ 7,800
X3-59	ツインビー	シャープ	X68000	5"2HD	¥ 7,800
X3-60	億万長者	コスモスコンピューター	X68000	5"2HD	¥ 9,800

注文No	タイトル	ソフトハウス	適応機種	メディア	価格
X3-61	戦国ソーサリアン	日本ファルコム	X1-ターボ	5"2D	¥ 3,800
X3-62	マスターオブモンスターズ	システムソフト	X-1ターボ	5"2D	¥ 8,000
X3-63	ラストハルマゲドン	ブレイングレイ	X-1シリーズ	5"2D	¥ 7,800
X3-64	リターンオブインター	SPS	X-1ターボ	5"2D	¥ 7,800
X3-65	スーパレイドッグ	T&Eソフト	X-1ターボ	5"2D	¥ 7,800
X3-66	ソーサリアン	日本ファルコム	X-1ターボ	5"2D	¥ 9,800
X3-67	イースII	日本ファルコム	X-1ターボ	5"2D	¥ 7,800
X3-68	マイト&マジック	スタークラフト	X-1ターボ	5"2D	¥ 9,800
X3-69	スーパー大戦略	システムソフト	X-1ターボ	5"2D	¥ 8,000
X3-70	アークス	ウルフチーム	X-1ターボ	5"2D	¥ 9,800
X3-71	パワフルまーじゃん	デービーソフト	X-1ターボ	5"2D	¥ 6,800
X3-72	白夜物語	イーストキューブ	X-1シリーズ	5"2D	¥ 7,800
X3-73	ファンタジーIII	スタークラフト	X-1ターボ	5"2D	¥ 9,800
X3-74	上海	システムソフト	X-1シリーズ	5"2D	¥ 6,500
X3-75	信長の野望全国版	光栄	X-1ターボ	5"2D	¥ 9,800
X3-76	三国志	光栄	X-1ターボ	5"2D	¥14,800
X3-77	ロードウォー2000	スタークラフト	X-1ターボ	5"2D	¥ 9,800
X3-78	ハイドライドIII	T&Eソフト	X-1シリーズ	5"2D	¥ 7,800
X3-79	マンハッタンレクイエム	リバーヒルソフト	X-1シリーズ	5"2D	¥ 7,800
X3-80	殺意の接吻	リバーヒルソフト	X-1シリーズ	5"2D	¥ 5,800
X3-81	ワールドゴルフII	エニックス	X-1ターボ	5"2D	¥ 7,800
X3-82	ソリテアロイヤル	ゲームアーツ	X-1シリーズ	5"2D	¥ 6,800
X3-83	まじゃべんちゃーねぎ麻雀	テクノポリスソフト	X-1シリーズ	5"2D	¥ 6,800
X3-84	大戦略マップコレクション	システムソフト	X-1ターボ	5"2D	¥ 4,800
X3-85	ディアブロ	ブローダバンドジャパン	X-1シリーズ	5"2D	¥ 6,800
X3-86	アルギースの翼	工画■スタジオ	X-1ターボ	5"2D	¥ 7,800

お申し込み方法

右の注文書にご希望商品の注文No.および必要事項ご記入の上、現金書留にて J&P 渋谷店までお申し込みください。現金受領後、発送いたします。
また、J&P HOTLINE会員の方は、ショッピングコーナーでもお申し込みいただけます。

●記載商品以外のご注文も承ります。
詳しくはお電話にてお問い合わせ下さい。
☎(03)496—4141 定休：毎週水曜日

キリトリ線

現金書留申込み用紙

おところ 〒□□□□□

TEL (　　　)

おなまえ　　　様

注文No	数量	金額
X3-　(　　)		円
X3-　(　　)		円
合計		円

お手持ちのパソコン

お申込み先：東京都渋谷区道玄坂2丁目28番4号(〒150) J&P 渋谷店メールショッピング係

■600%幅パソコンラック

X3-300
パソコンラック
シグマPW-833 ¥29,500
J&P特価 16,800円
W600×D640×H1280%

X3-301
¥32,800を J&P特価 15,800円
パソコンラック
エレコムDS-20 ¥32,800
W650×D700×H1260%
- ロック式キャスター付
- コード落しボックス付

X3-302
¥43,000を J&P特価 19,800円
パソコンラック
エレコムPD-02 ¥43,000
W640×D700×H1305%
- ロック式キャスター付 ●コード落しボックス付
- 中棚が2枚でキーボード収納棚又はペーパー置きにできます。
- 2Pコンセント付

X3-303
パソコンラック
エレコムPD-01 ¥45,000
J&P特価 39,800円
W650×D700×H1245%
- ロック式キャスター付 ●コード
- コード落しボックス付

X3-304
トール・ステーション・デスク
シグマPW-6502 ¥44,000
J&P特価 39,800円
W650×D650×H1345%
- ロック式キャスター付 ●パソコンチェアー、原稿台別売

X3-305
パソコンラック
エレコムDS-10 ¥32,000
J&P特価 16,800円
W650×D710×H1285%
- ロック式キャスター付
- コード落しボックス付
- コンセント付
- 2Pコンセント2個付

X3-306
パソコンラック
コニカED-50 ¥29,800
J&P特価 14,800円
W640×D700×H1260%
- ロック式キャスター付
- コードクランプ付

X3-307
パソコンラックER-600
オーバートップデスクER-606付
エレコムER-600
オプションプリンタ台ER-606
J&P特価 15,800円 ¥合計34,980
W650×D625×H1355%
- 2Pコンセント2個付

X3-308
パソコンラック
サンワSR-106 ¥36,800
J&P特価 17,800円
W620×D700×H1265%
- ロック式キャスター付 ●コード落しボックス付 ●コンセント付
- キャスター付

X3-309
ラックI model 2
サンワRAC-312 ¥38,800
J&P特価 36,800円
W662×D675(キーボード収納時580×H1186%
- キャスター付
- キーボード収納トレー付

■900%幅パソコンラック

X3-310
パソコンデスク
エレコムPD-99
J&P特価 29,800円 ¥48,000
W900×D700×H1280%
- ロック式キャスター付 ●コード落しボックス付 ●コンセント付 ●B4判引き出し別売¥3,500

X3-312
¥49,800を J&P特価 42,900円
ワーク・ステーション・デスク
シグマPW-9300 ¥49,800
W900×D740×H875
- コンセント、手許スイッチ付 ●5"フロッピィサイズ引出し ●データスタンド、ペーパーストレイジOAチェアー別売

X3-311
パソコンデスク
エレコムER-900
J&P特価 29,800円 ¥43,000
W900×D700×H1270%
- ロック式キャスター付 ●B4判引き出し付

X3-313
パソコンデスク
サンワSR-107
J&P特価 29,800円 ¥41,800
W940×D700×H1265%
- ロック式キャスター付 ●コード落し付
- 2Pコンセント2個付

■1,200%幅パソコンラック

X3-314
パソコンデスク
エレコムPD-120 ¥48,000
J&P特価 31,500円
W1200×D700×H820～1180%
- ロック式キャスター付 ●オーバートップ調節可 ●コンセント付 ●B4判引き出し別売 ¥3,500

X3-315
¥48,000を J&P特価 28,000円
パソコンデスク
エレコムER-1200 ¥48,000
W1200×D700×H820～1180%
- ロック式キャスター付
- オーバートップデスク高さ調節可

X3-316
98ステーションD-II
サンワDSF-992L ¥59,800
J&P特価 56,800円
W1200×D800×H840～1190%
- 手元集中スイッチコンセント付
- オーバートップ調節可 ●コード落し ●ブックエンド付

X3-317
パソコンデスク
エレコムEDX-1212 ¥69,800
J&P特価 62,800円
W1200×D800×H680(+460)%
- 手許集中スイッチコンセント付
- ロック式キャスター付 ●中棚は上下5段階調節可 ●オーバートップデスク付

X3-318
トレーユニット(オプション)
エレコムFO-50E ¥4,600
J&P特価 4,200円
W377×D270×H37%
B4判サイズで、よく使う原稿やプリント用紙を収納するのに便利です。

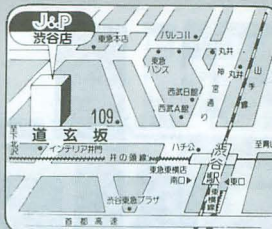
X3-320
98ステーションD-V
サンワDSF-995L ¥69,800
J&P特価 66,500円
W1200×D800×H840～1190%
- 手許集中スイッチコンセント付
- オーバートップ調節可
- コード落し ●ブックエンド付

X3-319
ペーパーストレイジ(オプション)
エレコムPO-90E ¥13,000
J&P特価 12,300円
15×11インチ連続用紙がスムーズにIN-OUTします。

¥はメーカー標準価格です。

全国無料配達

ピング

メールショッピングのお申し込みは J&P 渋谷店で承ります。

Personal Computer Store

J&P 渋谷店

東京都渋谷区道玄坂2丁目28番4号(〒150)
☎(03)496-4141〈水曜定休〉

■OAチェアー

¥12,000を J&P特価 **5,980円**

X3-321
OAチェアー
スター L-395 ¥12,000
●張地/布、ネジ式座面上下調節 ●キャスター付●色/ブルー、ブラウン、グレー

X3-322
OAチェアー
エレコムCCF-220 ¥20,000
J&P特価 **12,800円**
W530×D530×H760%
●張地/布、ネジ式座面上下調節 ●ヒジかけ、キャスター付

X3-323
OAチェアー
エレコムCCF-30 ¥32,000
J&P特価 **18,800円**
●張地/布、ガス式座面上下調節 ●キャスター付

X3-324
OAチェアー
シグマET-101F ¥26,500
J&P特価 **25,000円**
W460×D620×H730~810%
●張地/布、バネロッキング式背もたれ ●キャスター付5本脚●ハンドル式上下昇降

X3-325
OAチェアー
サンワSNC-087 ¥25,000
J&P特価 **23,800円**
W450×D550×H760~860%
●張地/布(ライトグレー)●バネロッキング式背もたれ●キャスター付4本脚●ハンドル式上下昇降

X3-326
バランスチェア(ブラック)
サンワ5064 ¥19,800
J&P特価 **18,800円**
●張地/布●前後に2段階調節できるシート●腰への負担が少なく、OA作業の能率UP

■その他のラック

ワープロユーザーにおすすめ!

X3-327
マルチラック
エレコムERX-7 ¥15,000
J&P特価 **12,800円**
W600×D700×H650%

X3-328
ホームデスク+チェアー
エレコムHMD-10-B
エレコムCCF-330-B ¥合計19,500
J&P特価 **12,800円**
W900×D445×H700%
●B4判引き出し3段付

■オプション

X3-331
机上パソコンラック
エレコムEK-30 ¥8,800
J&P特価 **6,580円**
W600×D400×H190%

X3-332
机上パソコンラック
エレコムEK-50W ¥16,800
J&P特価 **9,800円**
W1200×D350×H195%

X3-333
マウステーブル
エレコムMT-1、MT-2
J&P特価 **3,500円** ¥5,500
MT-1/対応機種PD-01、02
MT-2/対応機種DS-10、20、ER-600、900

ラップトップパソコンユーザーにおすすめ!

X3-329
パソコンラック
エレコムPD-500 ¥15,000
J&P特価 **12,800円**
W500×D625×H835%
ラップトップパソコンにピッタリ/門口500%サイズの省スペースラック

CADユーザーにおすすめ!

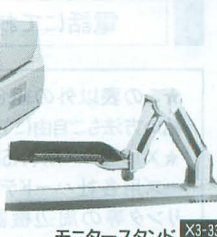
¥79,800を J&P特価 **74,800円**

X3-330
CADデスク Type I
サンワCAD-101 ¥79,800
W1000×D1000(天板収納時D700)×H855+1035%
●CPUボックス/アクリル戸付●キーボードスライダー付●コンセント3P・3個口付●手許集中スイッチ付

キーボードらくらく収納!

X3-334
キーボードドロワー
サンワTOK-020 ¥11,800
J&P特価 **11,200円**
W520×D400+260×H92%
手置台付(木製)

X3-335
キーボードドロワー
サンワYA-KB001 ¥9,800
J&P特価 **9,300円**
W630×D395+260×H100%
手置台付(アクリル製)

X3-336
デスクアダプター
エレコムAD-10
J&P特価 **19,000円** ¥20,000
W500×D700+268%
伝票台が付いていますので、業務がたいへんスムーズに行なえます。

X3-337
モニタースタンド
M.S.C. YU-M11 ¥29,800
J&P特価 **19,800円**
耐久重量60kg
14、15インチモニター用
机の上が広々と使えます。

お申し込み方法

右の注文書にご希望商品の注文No.および必要事項ご記入の上、現金書留にて J&P 渋谷店までお申し込みください。現金受領後、発送いたします。
また、J&P HOTLINE会員の方は、ショッピングコーナーでもお申し込みいただけます。

●記載以外のパーツのご注文も承ります。
詳しくはお電話にてお問い合わせ下さい。
☎(03)496-4141 定休:毎週水曜日

キリトリ線

おところ 〒□□□□□	注文No.	数量	金額
	X2-　(　)		円
	X2-　(　)		円
TEL (　)	合計		円
おなまえ 様	お手持ちのパソコン		

お申込み先:東京都渋谷区道玄坂2丁目28番4号(〒150) J&P 渋谷店メールショッピング係

クリエイト特典

- ●全商品完全保証書付(メーカー保証)
- ●全国無料配達(一部離島の方は有料になります)
- ●配達日の指定OK(日曜・祭日にかかわらずお客様のご都合にあわせて配達します)
- ●どんな商品の組合せも自由自在(ご予算、用途に応じ自由自在にシステムアップできます)
- ●中古パソコン高額下取り(今お使いのパソコンをわずかな差額でグレードアップ)
- ●お支払い方法自由(低金利の均等払い、ボーナス一括払いもご利用ください)

営業時間(年中無休)
AM10:00〜PM7:00(日曜・祭日はPM6:00まで)

当社はX68000の販売認定店です。どんなことでも安心してご相談ください。

(★X68000をお買上げのお客様にもれなく、▶X68000オリジナルテレホンカードプレゼント!!)

基本セット X68000 ACE

●CZ-601C(本体+キーボード)	¥319,800
●CZ-601D(カラー専用ディスプレイ)	¥119,800
●CZ-6ST1(チルトスタンド)	¥5,800
●ブランクディスケット(5"2HD・10枚)	¥10,000
●ソフト/アルカノイド	¥サービス
■定価合計	¥455,400▶クリエイト特価

電話にてお問合せください。

格安基本セット X68000 ACE

●CZ-601C(本体+キーボード)	¥319,800
●CZ-603D(カラー専用ディスプレイ)	¥84,800
●CZ-8PC3(熱転写カラー漢字プリンタ)	¥65,800
●ソフト/アルカノイド	¥サービス
●プリンタ用紙	¥サービス
●ブランクディスケット(5"2HD・10枚)	¥10,000
■定価合計	¥480,400▶クリエイト特価

電話にてお問合せください。

基本セット X68000 ACE HD

●CZ-611C(本体+キーボード)	¥399,800
●CZ-601D(カラー専用ディスプレイ)	¥119,800
●CZ-6ST1(チルトスタンド)	¥5,800
●ブランクディスケット(5"2HD・10枚)	¥10,000
●ソフト/アルカノイド	¥サービス
■定価合計	¥535,400▶クリエイト特価

電話にてお問合せください。

Vーセット X68000 ACE

●CZ-601C(本体+キーボード)	¥319,800
●CZ-601D(カラー専用ディスプレイ)	¥119,800
●CZ-8PC2(熱転写カラー漢字プリンタ)	¥65,800
●CZ-6TV1(カラーイメージユニット)	¥69,800
●CZ-6ST1(チルトスタンド)	¥サービス
●ブランクディスケット(5"2HD・10枚)	¥10,000
●ソフト/アルカノイド	¥サービス
■定価合計	¥585,200▶クリエイト特価

電話にてお問合せください。

ミュージックワークセット X68000 ACE

●CZ-601C(本体+キーボード)	¥319,800
●CZ-601D(カラー専用ディスプレイ)	¥119,800
●CZ-8PC3(熱転写カラー漢字プリンタ)	¥65,800
●SOUND PRO-68K(音色作成ツール)	¥15,800
●CZ-6ST1(チルトスタンド)	¥5,800
●ブランクディスケット(5"2HD・10枚)	¥10,000
●MUSIC PRO-68(楽譜入力ツール)	¥サービス
■定価合計	¥537,000▶クリエイト特価

電話にてお問合せください。

基本セット X1 turbo Z III

●CZ-888CBK(本体+キーボード)	¥169,800
●CZ-860DBK(カラーディスプレイ)	¥99,800
●CZ-6ST1(チルトスタンド)	¥5,800
●ブランクディスケット(5"2HD・10枚)	¥10,000
■定価合計	¥285,400▶クリエイト特価

電話にてお問合せください。

大サービス・ゲーマーズセット X68000 ACE

●CZ-601C(本体+キーボード)	¥319,800
●CZ-601D(カラー専用ディスプレイ)	¥119,800
●ドラゴンスピリッツ	¥8,800
●沙羅曼蛇	¥8,800
●XE-1 PRO(ジョイスティック)	¥9,800
●ドッジボール	¥サービス
●アルカノイド	¥サービス
●CZ-6ST1(チルトスタンド)	¥サービス
■定価合計	¥467,000▶クリエイト特価

電話にてお問合せください。

グラフィックワークセット X68000 ACE HD

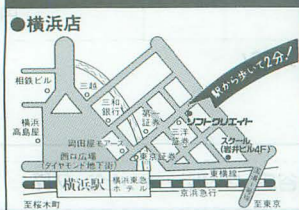
●CZ-611C(本体+キーボード)	¥399,800
●CZ-611D(0.31ピッチ・カラーディスプレイ)	¥145,000
●CZ-6PV1(カラービデオプリンタ)	¥198,000
●Z's STAFF PRO-68K	¥58,000
●レイトレーシングソフト	¥68,000
●CZ-6ST1(チルトスタンド)	¥サービス
●ブランクディスケット(5"2HD・10枚)	¥10,000
■定価合計	¥878,800▶クリエイト特価

電話にてお問合せください。

新品超お買得品セット

●CZ-820CE	¥69,800
●CZ-820DE	¥79,800
●CZ-503F(5インチシングルドライブ)	¥49,800
■定価合計	¥199,400

大特価¥78,800

★この表以外の組合せ、お支払い方法もご自由にできます。
★X1シリーズ用、X68000シリーズ用各社ハードディスク/プリンタ等の周辺機器を大特価にて販売しております。
電話にてお問合せください。

X68000シリーズ用 周辺機器お買い得セール

型番	品名	定価	特価
CZ-6VT1	カラーイメージユニット	¥69,800	クリエイト特価
CZ-8NS1	カラーイメージスキャナ	¥188,000	
CZ-6BE1A	1MB増設RAMボード	¥38,000	
CZ-6BE2	2MB増設RAMボード	¥79,800	
CZ-6BE4	4MB増設RAMボード	¥138,000	
CZ-6BU1	ユニバーサルI/Oボード	¥39,800	
CZ-6BG1	GP-IBボード	¥59,800	
CZ-6BP1	数値演算プロセッサ・ボード	¥79,800	

型番	品名	定価	特価
CZ-6EB1	拡張I/Oボックス(4スロット)	¥88,000	クリエイト特価
CZ-6BC1	FAXボード	¥79,800	
CZ-6BN1	スキャナ用パラレルボード	¥29,800	
CZ-8BS1	ステレオFM音源ボード	¥23,800	
CZ-603D	ドットピッチ0.31㎜ 14型高解像度	¥84,800	
CU-14CD	ドットピッチ0.31㎜ 14型高解像度	¥84,800	
CU-14ED	ドットピッチ0.39㎜ 14型高解像度	¥79,800	
AN-8TU	パソコンチューナ	¥35,800	

▲上記以外ビジネスソフト、最新ゲームソフト豊富に在庫あります。※送料はご注文の際お問合せください。

総合お問合せ先☎03-486-6541(代)

パソコン専門ショップ

ソフトクリエイト 渋谷/横浜

●渋谷店☎03-486-6541(代)
〒150:東京都渋谷区渋谷1-12-7 三和渋谷ビル
振込銀行:三井銀行 渋谷宮益坂支店(普)No.5000340

●横浜店☎045-314-4777(代)
〒221:横浜市神奈川区鶴屋町2-12-8 第1建設ビル
振込銀行:三和銀行 横浜駅前支店(普)No.310852

熱い心を伝えるミュージックテクノロジー

ソングファイル 68Kシリーズ登場

Musicstudio PRO-68K対応

佐久間正英ソングファイル SF-002 定価5,800円

プラスティックス再結成コンサートで話題を呼んだ佐久間正英は、"BOØWY""Street sliders""Blue Hearts"等のプロデュースをてがけています。ソロアルバム『LISA』などブイ・エフ・ブイスタジオを中心に独自の音楽活動を展開し、海外からも高い評価を得ています。

国本佳宏ソングファイル SF-001 定価5,800円

かつてサザンオールスターズにも参加していた国本佳宏は、現在スタジオで作詞、作編曲、エンジニアとトータルなサウンドをプロデュースしています。冨田勲のサウンドクラウドシリーズサポート、スティービーワンダーのアレンジ、サイトロンレーベルの音楽担当と、幅広く世界的な活動をしています。

Yoshihiro Kunimoto

SAN MUSICAL SERVICE
〒154 東京都世田谷区池尻4-1-4 TEL.03(419)8839

ソングファイル68Kは、Musicstudio PRO-68K(X68000)対応の"オリジナルデータ曲集"です。音色はMT-32(ローランド社製)に合わせています。パレットの上で色絵の具を混ぜ合わせるように、あなたの感性で音創りをしてください。

ROLAND MT-32

《広告の半ページ》月刊電脳倶楽部はただ今発売中でーす。

月刊 電脳倶楽部 89年3月号(Vol.10) 2月17日発売

2HDディスクに入ったX68000のための雑誌だっ!

恐怖と戦慄のプログラム：貴方の行為?に可愛く♡応える

げぼぼディスクドライブ

源平討魔伝の音声データを料理する

源平カッター

先月のは小手調べだっ! スペハリ改造キャラクタデータ第2弾
今月分はステージ2以降のデータです。
なお,第3弾で完結の予定です。

それからそれから
熱血パソ根小説『IBMの星』青春叙情編・「君の瞳は100万バイト」,好評連載『はるかカナダより』
その他いろんな記事,ツール,データ,ビープ音などを満載!

(内容は一部変更されることがあります。ご了承ください」

編集長祝一平からの御挨拶「というわけで.最近X68000の新機種についてあれこれガセネタが飛びかってますが,うん,元気があってよろしい」

満開製作所 電脳倶楽部 編集部
〒171 東京都豊島区要町1-3-24 三浦ビル3F
TEL.(03)554-9282/FAX.(03)554-3856

販売方法は通信販売のみです。お申し込みの方法は左記の住所へ現金書留で
定期購読 6ヵ月分 6,000円(郵送料サービス)
◆2月17日以降に受け付けた分は,原則としてVol.10から発送します。新たに購読を希望される方は,「新規」と御明記下さい。
◆郵便振替を御利用の場合は口座番号「東京5-362847 満開製作所」でお願いいたします。
製品の性格上,返品には応じられませんが,お申し出があれば定期購読を解約し残金をお返しします。
(なお,バックナンバーの受け付けは,春以降に再開の予定です)

ACCESS

X1エミュレータ

定価¥9,800

●X68000でX1を体験したい君に！

X1エミュレータはX68000上でX1シリーズのアプリケーションを実行するソフトエミュレータです。X68000上に実現した仮想X1マシンをお楽しみ頂けます。

●X1ソフトをX68000で遊びたい君に！

X1ソフトをX68000上にファイル転送できますので、これまでにX1で作った多くのプログラムをX68000で体験できます。

●やっぱりX1がかわいい君に！

X68000を使いながらもX1を使っている気持ちになれます。

実行可能アプリケーションソフト

●HuBASIC ●X1 CP/M ●X1 LOGO
【X1 CP/M用 ランゲージシリーズ】●APL ●LISP ●COBOL ●C ●FORTH ●FORTRAN ●PASCAL
●etc（X1シリーズ用とされているものに限ります。）

＊プロテクトの施してあるソフトは実行できません。
＊一部サポートしていない機能があります。
＊タイミング等ハードウェアに依存するようなものは、原理上実行できない、もしくは正常に動作しない場合がありますのでご注意ください。
＊実行速度はX1と比較して約3～5倍程度遅くなります。
＊turbo専用のソフトは動作致しません。

X1 5" 2D ⟷ X68000 Human68k

ファイル転送ユーティリティ

ディスク転送　X1ディスク↔X68000 Human68k（5" 2Dディスクイメージファイル）
X1エミュレータはHuman68kディスク上のX15" 2Dディスクイメージファイルを仮想ドライブとして使用！

ファイル転送　X1 BASIC↔X68000 Human68k：X1 CP/M↔X68000 Human68k
X1で作ったプログラム&データがX68000で使える！

メニューで実行　このユーティリティはメニューにそって実行するので操作は簡単！

＊ファイルを転送するために専用ケーブルが付属します。X1とX68000をつないでご使用ください。
★X1エミュレータの購入方法　このほど、このX1エミュレータは、直接弊社よりみなさまにお届けできるようになりました。詳しくはお問合せください。

MS-DOS エミュレータ

CONCERTO-X68K

定価¥99,800

CONCERTO-X68K（コンチェルト）はX68000上でお使い頂くMS-DOSエミュレータです。専用ハードウェア：DOS Engineとエミュレーションソフトで構成され、特定機種専用のものを除くMS-DOS V2.11のソフトがX68000上でお使い頂けます。DOS EngineはNEC V30 CPUを使用しており、MS-DOSソフトの高速実行を実現しております。1台のマシンで全く異なるハードをコントロール。X68000自身の持つ高速ディスクアクセス等の優れた性能をいかし、使い慣れたMS-DOSソフトをそのままご利用頂けます。これによりX68000の世界がさらに広がります。

専用ハード：DOS Engine

●8MHzのV30を使用（メモリノーウェイト）
●ボード上にMS-DOSの実行用メモリ512KByte搭載
●数値演算プロセッサ8087-1実装可能（オプション）
＊ボードは本体より12cm程度大きくなります。その部分にはカバーがつきます。

MS-DOS用実行可能アプリケーションソフト

●MS-C（Ver 3.00、4.00）
●MS-FORTRAN（Ver3.13、4.01）
●MS-PASCAL（Ver3.13）
●MS-LINK（Ver2.01、2.20、2.44）
●Lattice C（Ver2.12、3.10）
●Optimizing-C（Ver2.20F）
●TURBO PASCAL（Ver2.00B、3.01A）
●Plink 86（Ver1.46）　etc……
（実行可能ソフトの一例です。）

代理店募集

アクセスではこれらの製品の発売にあたり代理店を募集しております。詳しくはお問い合せください。

＊MS-DOSはマイクロソフト社、CP/Mはデジタルリサーチ社の商標です。
COMMAND.COMはMS-DOSに標準のコマンドプロセッサです。上記のソフトウェアは各社の商標です。
＊製品の仕様、名称は予告なく変更する場合もございますのであらかじめご了承ください。

有限会社アクセス　〒101 東京都千代田区神田神保町1-64 神保町協和ビル7F
☎03(233)0200(代)　FAX.03(291)7019

資料請求券 oh!X 3月号

ホットくんと
ラインちゃんの

パソコン通信のススメ

通信生活！始めます。

そのほか楽しいメニューがいっぱい！
HOT LINEはアメージング・ランドです。

このほかにも、ソフトのやり取りができるXMODEM機能や、居ながらにして買い物のできるオンラインショッピング、デイリーな株価の取り込み・分析（専用ソフト使用）をサポートしたCUGなど、便利で楽しい機能が満載。HOT LINEはまさにアメージング・ランドです。

■申込先
〒556 大阪市浪速区日本橋5-6-7 上新電機株式会社
J&P HOTLINE事務局宛 TEL(06)632-2521

■利用料金について
入会金／3,000円（スタータキット購入の代金から充当されます。）
接続料／3分あたり20円（アクセスポイントまでの電話代は含みません。）

スタータキット申込書

お名前		お電話番号	(　　　) －
ご住所	〒		

スタータキット（ID番号・パスワード 入会申込書・マニュアル同梱） ¥3,000

●パソコン/ワープロ通信ネットワークサービス
J&P HOT LINE

スタータキットのお求めは、下記のJ&P各店でどうぞ。

渋谷店	東京都渋谷区道玄坂2丁目28番4号☎(03) 496-4141	阪急三番街店	大阪市北区芝田1-1-3阪急三番街B1☎(06) 374-3311	さんのみや1ばん館	神戸市中央区八幡通3-2-16☎(078)231-2111
町田店	東京都町田市森野1丁目39番16号☎(0427)23-1313	高槻店	高槻市高槻町11番16号☎(0726)85-1212	京都寺町店	京都市下京区寺町通仏光寺下ル恵美須之町549☎(075)341-3571
八王子店	東京都八王子市旭町1番1号八王子そごう7F☎(0426)26-4141	くずは店	枚方市楠葉花園町15番2号☎(0720)56-8181	京都近鉄店	京都市下京区烏丸通七条下ル東塩小路町702☎(075)341-5769
テクノランド	大阪市浪速区日本橋5丁目6番7号☎(06) 634-1211	千里中央店	豊中市新千里東町1-3-204千里サンタウン3F☎(06) 834-4141	姫路店	姫路市東延末1丁目1番住友生命姫路南ビル1F☎(0792)22-1221
メディアランド	大阪市浪速区日本橋5丁目8番26号☎(06) 634-1511	摂津富田店	高槻市大畑町24－10☎(0726)93-7521	和歌山店	和歌山市元寺町4丁目4番地☎(0734)28-1441
コスモランド	大阪市浪速区難波中2丁目1番17号☎(06) 634-3111	寝屋川店	寝屋川市緑町4－20☎(0720)34-1166	奈良1ばん館	奈良市三条町478－1☎(0742)27-1111
ワーフロランド	大阪市浪速区日本橋4丁目9番15号☎(06) 634-1411	藤井寺店	藤井寺市岡2丁目1番33号☎(0729)38-2111	西宮店	兵庫県西宮市河原町5－11☎(0798)71-1171
ビジネスランド	大阪市北区梅田1-1-3大阪駅前第3ビルB2☎(06) 348-1881	岸和田店	岸和田市土生町2451－3☎(0724)37-1021	郡山インター店	大和郡山市横田693－1☎(07435)9-2221

ADVANCED TURBO

先駆の"Z"アビリティがパソコンクリエイターを魅了する。

新登場

パソコンテレビ

X1 turbo Z III

パーソナルコンピュータ+キーボード+マウス	CZ-888C-BK	標準価格 169,800円
14型カラーディスプレイテレビ	CZ-860D-BK	標準価格 99,800円
チルトスタンド	CZ-6ST1-B	標準価格 5,800円

クリエイティブマインドを刺激するAV機能　テレビ、ビデオ、ビデオディスクなどの映像を最大4,096色のリアルな画像で瞬時にグラフィック画面に取り込めるカラー画像デジタイズ機能を標準装備。4段階の量子化取り込み、42通りのモザイク取り込みなど多彩なトリック取り込み処理もサポート。さらにクロマキー合成、インターレーススーパーインポーズ、4,096色対応デジタルテロッパ機能、ステレオFM音源…先駆のAV機能がアートワークの領域をさらに拡げます。

AV指向の高水準ベーシックZ-BASIC搭載　多色グラフィック、カラー画像処理、ステレオFM音源、バンクメモリ対応など、ターボZシリーズが本来もつクリエイティブな機能をフルサポート。また豊富な画面モードで多色を駆使するときに便利なグラフィック用関数(HSV, RGB, HALF, CDOWN, CUP)も装備。さらにFM音源制御用ステートメントとしてX68000と命令コンパチの拡張MMLの採用によりスムーズな8音同時演奏を実現しています。

●メインメモリ128Kバイト標準装備、Z-BASICで最大576Kバイトまでサポート●1Mバイトの5インチフロッピーディスクドライブ2基搭載●JIS第1/第2水準準拠漢字、「システム・ユーザー辞書」を標準装備した高度な日本語処理機能●ニューデザインのマウス標準装備●X1ターボシリーズの豊富なソフト資産が活用できるコンパチブル設計●プリンタ、RS-232Cなど豊富なインターフェイスを装備●ドットピッチ0.39㎜のハイコントラストブラウン管、15kHz/24kHzのデュアルスキャン方式採用14型カラーディスプレイテレビ(別売)。

資料請求券 X1turbo ZIII Oh!X 3係

シャープ株式会社　●お問い合わせは…シャープ㈱電子機器事業本部システム機器営業部 〒545 大阪市阿倍野区長池町22番22号 ☎(06)621-1221(大代表)
電子機器事業本部テレビ事業部第4商品企画部 〒162 東京都新宿区市谷八幡町8番地 ☎(03)260-1161(大代表)

T4910217903543　雑誌02179-3

Oh!X 3月号 1989年3月1日発行(毎月1回1日発行) 通巻第83号 昭和58年11月2日第三種郵便物認可 ㈱日本ソフトバンク発行 Printed in Japan 定価540円